A Deep Learning Pipeline for Classifying Different Stages of Alzheimer's Disease from fMRI Data

Yosra Kazemi

Submitted in partial fulfilment
of the requirements for the degree of

Master of Science

Department of Computer Science
Brock University
St. Catharines, Ontario

Abstract

Alzheimer's disease (AD) is an irreversible, progressive neurological disorder that causes memory and thinking skill loss. Many different methods and algorithms have been applied to extract patterns from neuroimaging data in order to distinguish different stages of AD. However, the similarity of the brain patterns in older adults and in different stages makes the classification of different stages a challenge for researchers.

In this thesis, convolutional neuronal network architecture AlexNet was applied to fMRI datasets to classify different stages of the disease. We classified five different stages of Alzheimer's using a deep learning algorithm. The method successfully classified normal healthy control (NC), significant memory concern (SMC), early mild cognitive impair (EMCI), late cognitive mild impair (LMCI), and Alzheimer's disease (AD). The model was implemented using GPU high performance computing. Before applying any classification, the fMRI data were strictly preprocessed to avoid any noise. Then, low to high level features were extracted and learned using the AlexNet model. Our experiments show significant improvement in classification. The average accuracy of the model was 97.63%. We then tested our model on test datasets to evaluate the accuracy of the model per class, obtaining an accuracy of 94.97% for AD, 95.64% for EMCI, 95.89% for LMCI, 98.34% for NC, and 94.55% for SMC.

Acknowledgements

First, I would like to thank my Supervisor Professor Sheridan Houghten of the Computer Science department at Brock University. Whenever I had a difficulty about my research or writing she was always there to help. She always encouraged me to develop my own ideas and guided me in the right direction whenever she thought I needed it.

I would also like to acknowledge Professor Brian J. Ross and Professor Ke Qiu of the Computer Science department at Brock university as my committee members, and I am very grateful to them for their very insightful comments.

Last but not least, I must express my very special gratitude to my parents and to my sister for supporting me spiritually and encouraging me throughout my years of study and writing this thesis. Thank you for all your support.

Yosra Kazemi

Contents

List of Tables

List of Figures

Chapter 1

Introduction

In this chapter, we will first discuss the problem we study in this thesis, and then describe the structure of the thesis.

1.1 Problem Statement

Alzheimer's disease (AD) is an irreversible, progressive neurological disorder that causes memory and thinking skill loss. The disease is a neurodegenerative type of dementia that begins with mild deterioration and gets progressively worse [43][58]. Reports show that more than 5 million Americans are living with Alzheimer's[2].

Alzheimer's disease progresses at various rates and each individual may experience different symptoms at different times [2]. Distinguishing different stages of disease is usually a challenge for researchers because the between-class variance in different stages of Alzheimer's disease is low. Thus, Scientists have become interested in studying the brain changes in Alzheimer's disease in order to have a better understanding of different stages and the changes that cause the disease.

Many neuroimaging tools and examinations are available to study the brain. Some of the most common brain imaging tools are magnetic resonance imaging (MRI), functional magnetic resonance imaging (fMRI), and positron emission tomography (PET) scan.

Machine learning methods have helped computer scientists to classify and distinguish different stages of Alzheimer's disease using neuroimaging data. However, distinguishing between different stages of Alzheimer's is not easy. Most previous works have classified the neuroimaging data into binary classes (AD vs. normal control) or into three classes (AD, mild cognitive impair, and normal healthy control). Different machine learning methods have been applied for classification, such as conventional

methods like the general linear model (GLM)[41], and multi-voxel methods such as support vector machines (SVMs) [14][8][13]. These will be discussed in Chapter 3.

In the case of using fMRI scans as a diagnostic method for understanding brain changes and functionality in different areas, we need a very careful classifier. The typical machine learning approaches contain a feature selection step. However, in deep learning feature selection is an automatic process. This feature of the deep learning algorithm has caused a significant improvement in the accuracy of learning methods. Besides, it will remove the level of subjectivity (selecting the feature to use) from the previous methods [48].

In this study, previously successful deep learning methods [48][52] are used to classify more classes of Alzheimer's disease. Our classifier in this study is able to classify five different stages of the disease: normal control (NC), significant memory concern (SMC), early mild cognitive impair (EMCI), late mild cognitive impair (LMCI), and Alzheimer's disease (AD). Classification of different stages of the disease is challenging work since it is hard to distinguish these classes. We used the deep learning model to classify our large dataset into different stages, and to the best of our knowledge we are the first to do so.

1.2 Thesis Structure

The remainder of this thesis is organized as follows.

Chapters 2 and 3 consist of the background information, relevant concepts and previous work. First, we are concerned with an introduction to Alzheimer's disease, fMRI scanning, and deep learning algorithms in Chapter 2. Chapter 3 gives an overview of the application of machine learning models such as deep learning to neuroimaging data.

Chapters 4 and 5 are the main part of this study. In Chapter 4, we describe our dataset, data acquisition and the preprocessing of the data. Chapter 5 describes the deep learning model (AlexNet) that we applied on our dataset. We also describe the experiment we design for training our dataset in Chapter 5.

Chapter 6 presents experimental results from using the AlexNet Architecture in GPU systems. The results are presented in two sections: the accuracy and loss of the validation datasets and the accuracy of the subject level on the test datasets.

Chapter 7 is the conclusion of our thesis and also includes a brief discussion of possible future work.

Chapter 2

Background

In this chapter, we will discuss briefly Alzheimer's disease and the changes in the brain that are caused by Alzheimer's. Then we will describe different ways of categorizing the stages of Alzheimer's. In the second sub-section, fMRI scanning will be discussed, including how the fMRI data is different from MRI data. The last part of this chapter is a general explanation of deep learning algorithms and different convolutional neural network architectures.

2.1 Alzheimer's Disease

Alzheimer's disease (AD) is an irreversible, progressive neurological disorder that causes memory and thinking skill loss and cognitive decline. The disease is a neurodegenerative type of dementia that begins with mild deterioration and gets progressively worse [43][58]. "Alzheimer's disease is the sixth leading cause of death in older people in the United States and more than 5 million Americans are living with Alzheimer's"[2].

Alzheimer's disease is known as the main cause of dementia [2]. Damage to the brain cells can cause dementia. As a result of this damage, brain cells are not able to communicate with each other properly, which causes the loss of cognitive functioning and behavioral abilities of the person. Dementia has different stages, from the mildest stage when the person starts to lose some cognitive functioning, to the most severe stage when the person is completely dependent on other people for living [2][43].

Different parts of the human brain are responsible for different functions (e.g. thinking, movement, memorizing, etc). Depending on which region of brain cells are damaged, the functionality of that region will be affected. Depending how the brain structure is changing, the causes of dementia can vary. Different types of dementia include Alzheimer's disease, Lewy body dementia, frontotemporal disorders, and vascular de-

mentia [2]. In Alzheimer's disease, changes in certain protein levels in the brain's cells affect the ability of neurons to communicate in the hippocampus region. The hippocampus is responsible for learning and memory in the brain and it is more likely for the brain's cells to be damaged in this area first. That is why an early sign of Alzheimer's disease is memory loss [2][43].

Alzheimer's disease is named after Dr. Alois Alzheimer. In 1906 he examined the brain of a woman who had died of a mental illness which affected her memory, language, and caused unpredictable behavior. He noticed abnormal clumps and tangled bundles of fibers in the patient's brain, which are now called amyloid plaques and neurofibrillary or tau tangles, respectively. These are now regarded as some of the main symptoms of Alzheimer's disease [58][43].

2.1.1 Brain Changes Associated with Alzheimer's Disease

Inside each healthy adult human brain there are about 100 billion neurons, with branching extensions that enable them to communicate with one another. The connections between neurons are called synapses, which release and detect chemicals to help the flow of information between neurons. The brain contains about 100 trillion synapses. These synapses let the signals transmit through the brain's neuronal circuits to create the cellular basis of memories, thoughts, sensations, emotions, movements and skills[2].

As mentioned before, increasing the level of the abnormal form of protein tau (tau or neurofibrillary tangles) inside the neurons and increasing the level of protein beta-amyloid (beta-amyloid plaques) outside of neurons are two of the main changes in a brain affected by Alzheimer's disease. Beta-amyloid plaques interrupt the communication between neurons and cause cell death while tau tangles block the reception of the nutrients and other essential molecules inside neurons. Because of these changes, the brains of people with severe Alzheimer's disease shrink due to cell loss. In brain scans one can see the widespread debris and inflammation inside the brain as a result of the dead neurons. Figure 2.1 shows the comparison between a normal healthy brain and a brain with Alzheimer's.

Researchers believe that the brain changes in Alzheimer's disease begin 20 or more years before the symptoms appear [2]. It is difficult to notice any symptoms when the initial changes are happening, because the brain is attempting to correct them so the individual can function normally. However, with increasing neuronal damage, the brain is not able to correct them and patients start showing some cognitive decline. As the amount of damage increases the patients show more obvious cognitive decline until the

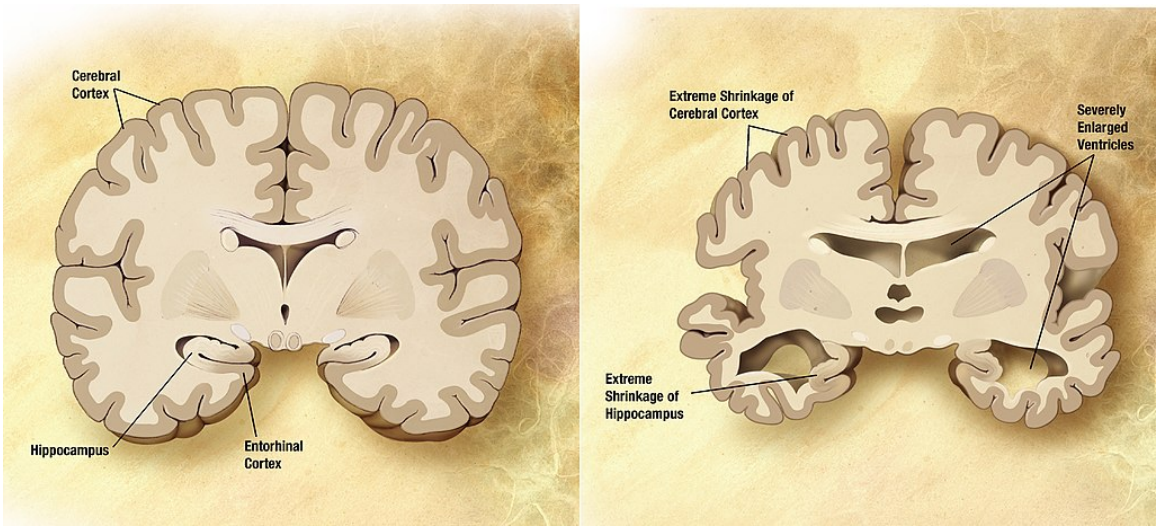


Figure 2.1: Comparison between normal brain (left) and brain of a person with Alzheimer's disease(AD) (right) [68].

patients are unable to have basic bodily functions such as moving their hands, walking or even swallowing.

2.1.2 Different stages of Alzheimer's disease

Changes in the brain begin many years before any symptoms of Alzheimer's disease appear in the patient. This first stage is referred to as pre-clinical Alzheimer's disease which can last for many years. The rate of progression of the disease through the next stages increases rapidly. In each stage patients show different symptoms and the disease may progress at different rates. There is no precise number of stages. Some experts use a seven stage model to better understand the progression of the disease [43][2]:

- Stage 1: No Impairment
- Stage 2: Very Mild Decline
- Stage 3: Mild Decline
- Stage 4: Moderate Decline
- Stage 5: Moderately Severe Decline
- Stage 6: Severe Decline
- Stage 7: Very Severe Decline

Alzheimer's disease is an individual experience. For each individual the symptoms and progression of the disease is different. Thus, it can be difficult to distinguish between the seven stages. The Alzheimer's association [43] generalizes the seven stages into three main categories: mild Alzheimer's disease, moderate Alzheimer's disease and severe Alzheimer's disease. However in terms of machine learning, what makes this classification a challenge is the low inter class variance. The goals of most machine learning feature extraction methods is to maximize the inter-class variance or minimize the intra class variance.

In this dissertation, we used 5 different categories, based on available subject data obtained from The Alzheimer's Disease Neuroimaging Initiative (ADNI) [1]. The Alzheimer's Disease Neuroimaging Initiative (ADNI) has a different way of categorizing the different stages of disease for the participants. Table 2.1 shows the ADNI study categories. In this categorization, healthy normal control corresponds to stage 1 (no impairment) in the previous categorization, Significant Memory Concern (SMC) could be considered as stages 2 and 3, Early Mild Cognitive Impairment (EMCI) as stage 4, Late Mild Cognitive Impairment (LMCI) as stage 5, and Alzheimer's disease (AD) is considered as stage 6 in the seven stages model.

Stages	Abbr.	Description
Cognitively Normal	CN	CN participants are the normal aging subjects with no signs of depression, mild cognitive impairment or dementia.
Significant Memory Concern	SMC	SMC participants considered in normal range of cognition. However, they show slight of memory concern. This is not consider as progressive memory impairment.
Early Mild Cognitive Impairment	EMCI	MCI participants are subjects with memory concern but no significant levels of impairment. They are able to do daily life activities and there are no sign of dementia. Levels of MCI (early or late) are determined by a neuropsychological test (Wechsler Memory Scale).
Mild Cognitive Impairment	MCI	
Impairment	LMCI	
Alzheimer's disease	AD	AD participants have been evaluated and meet the NINCDS/ADRDA criteria for probable AD

Table 2.1: Description of different stages of ADNI participants[1].

2.1.3 Diagnosis of Alzheimer's Disease

Diagnosing Alzheimer's disease requires a complete and exhaustive medical test. It is quite easy for physicians to determine if a patient has dementia, but it is difficult to identify the causes of the dementia. It is a long process for the physicians to make their diagnosis based on patient tests and examination results. There are many ways to diagnose Alzheimer's disease. Physicians, with the help of neurologists and geriatricians, use different approaches and tools to make a diagnosis. Some of these approaches and tools are as follows:

- Patients' medical and family history, including psychiatric and cognitive history.
- Observing patients' changes in behavior and thinking skill.
- Conducting a Mini Mental State Examination (MMSE) and physical and neurobiological exams.
- Blood test and brain imaging to determine other dementia causes, including vitamin deficiencies or tumors.

There are many different neuroimaging techniques to help physicians have a better understanding of the patient's progress, and make a more accurate diagnosis. One of the most used techniques is magnetic resonance imaging (MRI) as an essential part of the clinical assessment. Studying a patient's MRI is now a valid biomarker to diagnose especially in the prodromal stages (i.e. the mild cognitive impairment (MCI) stage). MRI measurements of atrophy show cumulative neuronal damage in the brain as one of the main reasons in the clinical state. In comparison with other biomarkers, cerebral atrophy has a strong correlation with cognitive decline [29]. However, some of the unusual types of AD have irregular atrophy patterns and in some cases atrophy patterns may have overlap with other diseases. Structural MRI is not able to specify the molecular behavior, nor to directly identify the other AD causes such as amyloid plaques and tau tangles. Therefore, new techniques have become available in the neuroimaging field, such as positron emission tomography (PET) and functional magnetic resonance imaging (fMRI) [29][49].

PET is a tool to measure metabolic and neurochemical processes in vivo by using radio-labeled ligands. There are two types of PET techniques being applied in Alzheimer's Disease research, fluorodeoxyglucose (FDG) to measure brain metabolism [24], and amyloid tracers [64][69].

In this dissertation, we are using fMRI imaging data, which measures brain activity during a cognitive task or experiments at rest by measuring blood oxygen levels and

blood flow. In fMRI studies, the blood oxygenation level dependent (BOLD) signal is an indicator of activity, and is calculated by changes in local blood flow and oxygenation level[44].

2.2 fMRI Scanning

The application of functional neuroimaging has grown significantly in recent years. Specifically, fMRI has recently taken on a dominant role in the field of neuroimaging as it is a non-invasive technique with great spatial and temporal resolution [35].

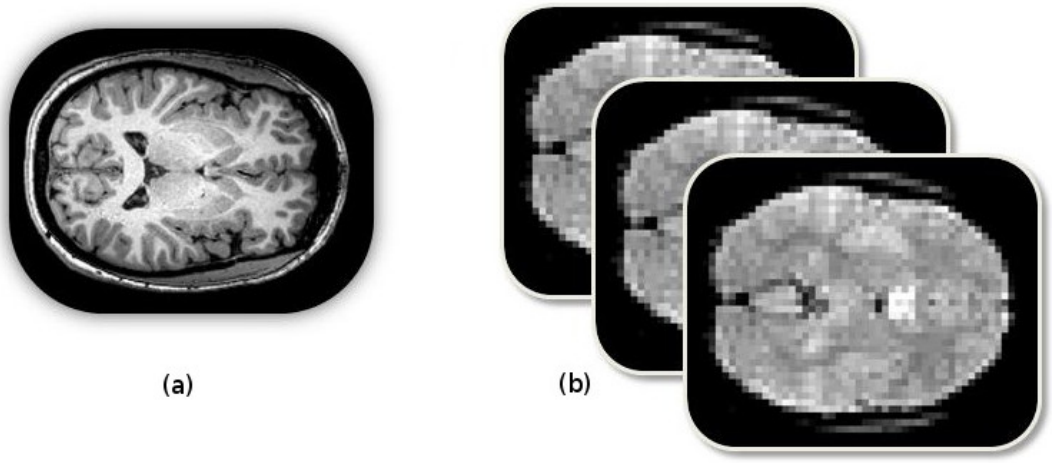


Figure 2.2: (a) is a single 3D volume of MRI with high resolution. (b) is a series of low resolution 3D volumes over time (i.e 4D data) of fMRI [4].

As previously mentioned, fMRI techniques provide information based on blood flow and oxygen level changes through neuronal activities. During an fMRI experiment, a series of images are obtained while the subject is either at rest, referred to as resting state fMRI (rs-fMRI), or performing some cognitive tasks (task based fMRI). The fMRI data contains a series of MRIs each of which consists of a number of voxels or spaced volume elements which divide the brain into boxes of the same size. In other words each fMRI image consists of several 3D MRI volumes which have been taken over time, for example every 1-4 seconds, and creates a 4D image of the whole brain (Figure 2.2).

In task based fMRI, subjects are asked to lie down in the scanner and perform some cognitive, sensory, memory or motor task. During the experiment the MRI scanner tracks the signal in the brain. In the areas responsible for the stimulus one would expect changes in the signals there over subsequent MRI images. These changes over time

can be localized in each voxel[19].

However, in this research we are using resting state fMRI imaging to classify the difference between Alzheimer's disease stages. Our focus is on how the brain is changing through different stages of the disease, not on how it is changing during an activity. Specific information about the dataset that we used will be described in Chapter 4.

2.3 Deep Learning and Image Classification

Machine learning (ML) approaches play a significant role in today's technology challenges. In many aspects such as web search, social networks, e-commerce, design and production of modern cars, cameras, smart phones and many other areas, machine learning extends human abilities. Machine learning techniques are used in image classification, object detection, language processing, and speech recognition.

Machine learning algorithms use two phases to classify information such as images. Figure 2.3 shows the machine learning phases for classification. The first step is the training phase, where the machine learning algorithm is trained with the input images and their corresponding labels. In machine learning classification, feature extraction is an essential step to extract new features from images for the training algorithm. Some of the main examples of the feature extraction algorithm in image classification are the scale-invariant feature transform (SIFT) and the histogram of oriented gradients (HOG). The second step is the prediction (test) phase, which uses the trained classifier in the previous phase to predict the labels of test images (previously unseen images). In this phase the same feature extraction algorithm is applied to the new images and the features are transferred to the trained model to predict the label [53] [16].

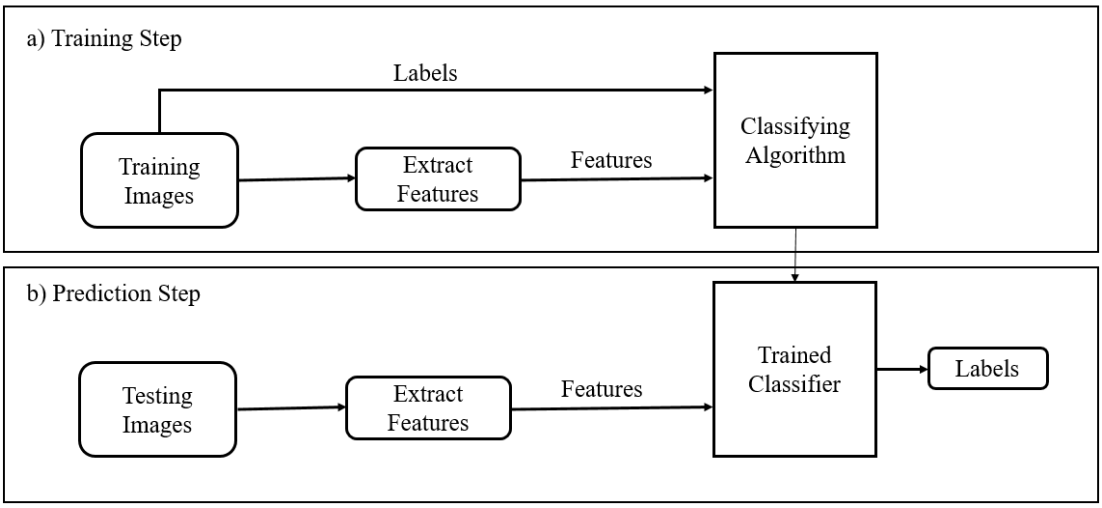


Figure 2.3: Two phases of machine learning algorithm for classifying images

For larger datasets researchers often use a new class of machine learning called deep learning[53] [16]. Deep learning is a class of artificial neural networks (ANNs) with many processing layers. However, for many years researchers' work on training deep architectures of ANNs was not successful until in 2006 with Geoffrey Hinton's research on "A Fast Learning Algorithm for Deep Belief Nets" [21] and "Reducing the Dimensionality of Data with Neural Networks" [22]. In addition to this cutting edge research, which was a response to the need for faster and more accurate algorithms for big datasets, the innovation and advancements in computing ability by GPUs was another main factor in the wide-spread use of deep learning.

The main difference between traditional machine learning models and the deep learning model is automatic feature extraction in the deep learning algorithm; this plays a significant role in improving the accuracy in different problems [48]. Another feature that distinguishes deep learning from other machine learning techniques is the depth of the model.

2.3.1 Artificial Neural Networks (ANNs)

As the artificial neural network term implies, these models are a class of machine learning algorithms that are inspired by the human brain's neural network systems. The human brain, as a command center for the human nervous system, consists of billions of neurons which are connected with about 10^{14} synapses. Each neuron, as a computational unit of the brain, has a cell body, dendrites and an axon. Each neuron receives and processes input signals from its dendrites and sends out the output signals through

its axon. Figure 2.4(a) shows the biological neuron structure of the human brain. In a mathematical model 2.4(b), signals (S_i) based on synapses strength (W_i) multiplicatively interact ($S_i W_i$) with dendrites of other neurons. Using this model one can show how synapse strengths (weights) can learn and control the influence of one neuron on other ones. Figure 2.4 shows all signals from dendrites traveling to the body cell where all incoming signals are summed. If the value of the sum is greater than a specific threshold, then neurons can fire and send a signal spike through the axon.

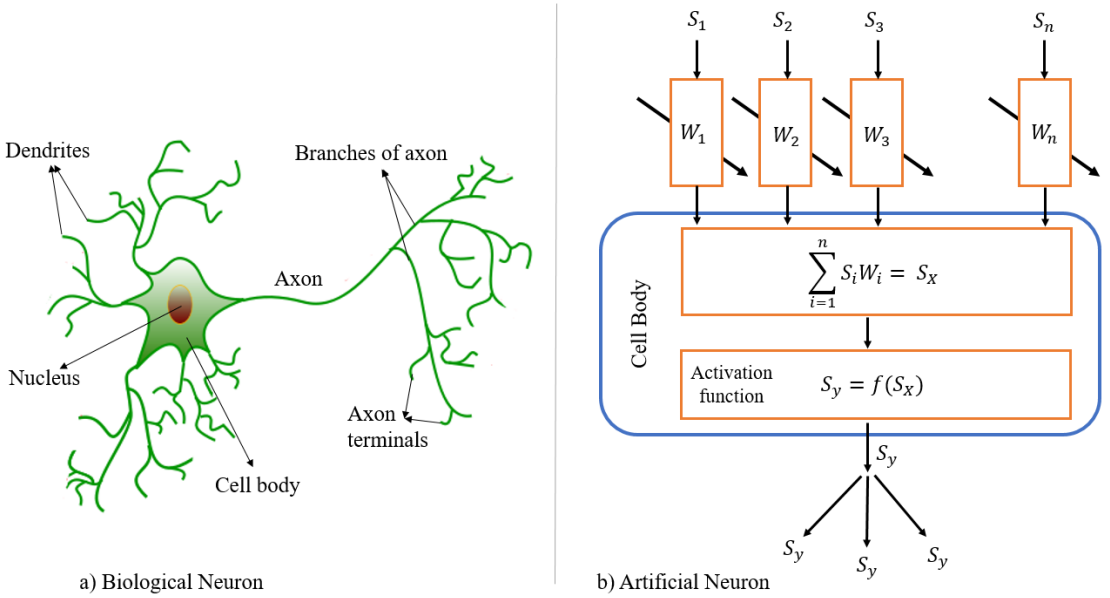


Figure 2.4: Drawing of biological neuron (left) and its mathematical model for artificial neuron (right) [11]

In an artificial neural network used to model biological neurons, we assume that only the frequency of firing is responsible for transmitting information while the exact timing of spikes is not a factor. The activation functions f are applied to model the firing frequency. Activation functions (also called transfer functions) are a way to represent the frequency of the spikes through the axon [39][7].

Figure 2.4(b) is representing a simple artificial neuronal network structure in which each artificial neuron has some inputs with their assigned weights, and an activation function. The activation function is applied to the weighted sum of inputs to produce the output of the system. Artificial neural networks are a connection between all of these artificial neurons.

In this model, the activation functions are non-linear in order to account for the complexity of the data. The most commonly used activation functions are:

- Sigmoid: $f(x) = \frac{1}{1+e^{-x}}$
- Hyperbolic tangent: $tanh(x) = \frac{2}{1+e^{-2x}} - 1$
- Rectified linear unit (ReLU): $f(x) = \begin{cases} 0 & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$

In deep neural networks the most common activation function is ReLU. On large and complex datasets, ReLU performs more effectively and faster than other activation functions [18].

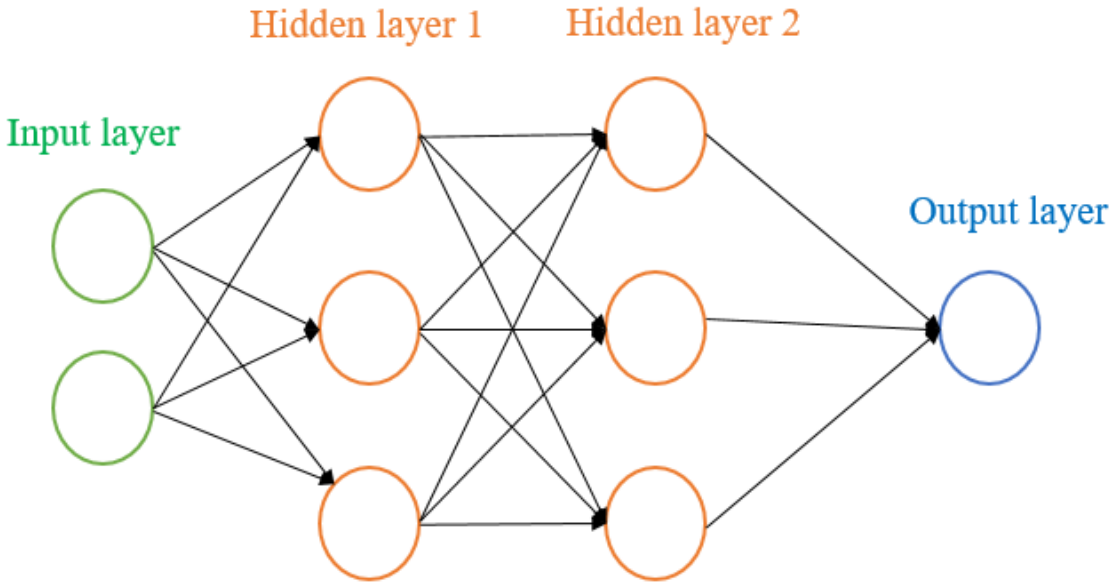


Figure 2.5: Feed-forward neural network with two hidden layers

Neural network models are comprised of distinct layers of neurons. The most common layer type is the fully-connected layer in which each node is connected to all of the next layer's nodes. One of the simplest topologies of artificial neural networks is feed-forward neural networks. These networks consist of three types of layers (input, hidden, and output layers). The free parameters in these models are the number of hidden layers and their size. In theory, we can model more complex problems with larger and deeper hidden layers. Figure 2.5 is an example of a fully connected feed-forward neural network with two hidden layers each with three neurons, an input layer of size two, and an output layer with one neuron.

For training neural network topologies, the goal is to learn the network's weights. For this purpose we need training data, which in image classification problems is a dataset

of images with their assigned labels. The second element is the loss function on the neuron's output to measure the inaccuracy of predictions. Then we are able to train the network using the backpropagation algorithm and gradient descent.

2.3.2 Convolutional Neural Networks (CNNs)

Convolutional neural networks (CNNs), are type of feed-forward networks that are inspired by the human visual cortex. CNNs use spatial relationships to reduce the number of learnable parameters. This feature in CNNs improves the feed-forward and backpropagation algorithms. The topology of CNNs has two main layers convolutional and pooling layers. these help the network to encode specific properties of images.

Convolutional Layer (CONV layer)

The convolutional layer is the most important block in CNNs' topology. In CNNs, the inputs to the lowest layer in the hierarchical architecture comes from small portions of the images (local receptive fields). These local receptive fields allow the processing units (neurons) to have access to the elementary features, including corners and oriented edges. This provides a consistent layer to shift, rotate, and scale. The convolutional layer is formed by neurons with learnable weights and biases (filters with learning abilities) that are spatially going over the input images. The convolutional layer computes the dot product of the entries of the filter and the connected region in the input volume. This produces a 2D feature (activation) map of the filter. Each filter will be active when they see the same particular feature in some spatial position on the image (input volume). All of these feature maps for all of the filters create the output volume [62]. For example, if a filter with a size of 5×5 is applied over an RGB (three color channels: Red, Green, Blue) image with a size of 32×32, to cover the RGB features of the image the filter must have a depth of three ($5 \times 5 \times 3$). The CONV layer accepts any volume of size $W_1 \times H_1 \times D_1$ (where W is the weight, H is the height, and D is the dimension) as an input. The size of the input volume, as well as four other hyper parameters are needed in CONV layer. They are the number of filters (K), filter's spatial extend (F), the size of stride (S) which controls how the filter convolves around the input, and the size of zero padding (P) which pads the input with zeros around the border. The outputs of the convolutional

layers are new volumes, ($W_2 \times H_2 \times D_2$), which are calculated as follows [51][17]:

$$\begin{aligned} W_2 &= \frac{(W_1 - F + 2p)}{S} + 1 \\ H_2 &= \frac{(H_1 - F + 2p)}{S} + 1 \\ D_2 &= K. \end{aligned} \tag{2.1}$$

Pooling Layer

The pooling layer performs as a non-linear down-sampling through the spatial dimensions. In CNN architectures, pooling layers are usually between convolutional layers. In the pooling layer the spatial size of the representation is reduced (down sampled) to minimize the amount of hyper parameters in the network, and as a result of down sampling this layer can control overfitting. For implementing the pooling layer many functions are available. However, the most common one is max pooling. Pooling operates on every depth slice of the input images and spatially resizes them [3][62]. The pooling layer accepts a $W_1 \times H_1 \times D_1$ volume and generates the new volume of size $W_2 \times H_2 \times D_2$ by the following formula [17]:

$$\begin{aligned} W_2 &= \frac{(W_1 - F)}{S} + 1 \\ H_2 &= \frac{(H_1 - F)}{S} + 1 \\ D_2 &= D_1. \end{aligned} \tag{2.2}$$

Where F is the spatial extent and S is the stride. In practice there are two most commonly used pooling layers. The first one is when $S = F = 2$ and the other pooling layer is when $S = 2$ and $F = 3$ which is called overlapping pooling [17]. AlexNet architecture uses overlap pooling [30].

Fully connected layer

Fully connected (FC) layers take the output of the last pooling layer and convert it to another space by non-linear methods. The most commonly used activation function in fully connected layers is hyperbolic tangent (tanh). Compared to other layers, fully connected layers have the most weights and therefore the training time in these layers increases consequently [17].

Convolutional Neural Networks Architecture

CNN architectures also include other network structures such as the normalization layer (ReLU layer) which uses a pairwise activation function [3]. The other layer is a fully connected (FC) layer which is responsible for calculating the class scores[62]. The architecture of CNNs often begins with an input followed by the series of convolutional and pooling layers. Normalization layers are inserted after convolutional layers and the architecture will end with fully-connected layers. Figure 2.6 shows an example of a simple architecture of a CNN. The network accepts an input image to a series of convolutional layers which are followed by pooling layers, and it ends with fully-connected layers.

The early layers in CNNs' architecture encode the elementary and general features of the images, while the later layers are responsible for encoding more detailed patterns in the input images. The CONV, pooling and ReLU layers are feature extractors, while the fully connected layers are classifiers in the model.

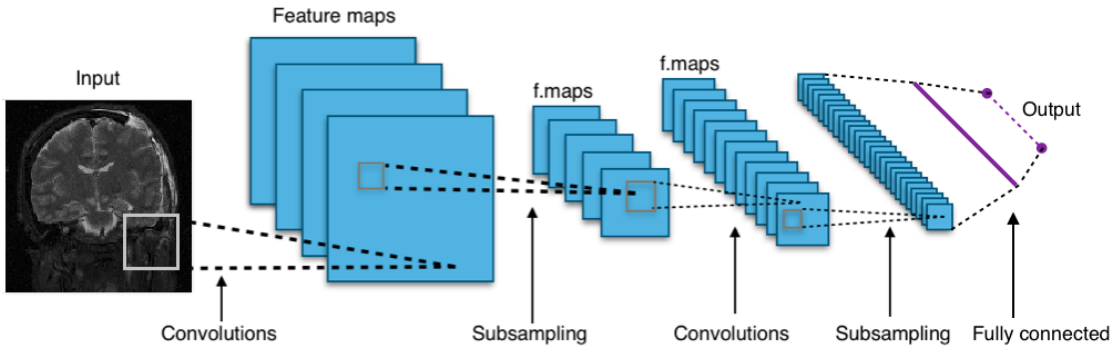


Figure 2.6: A simple CNN architecture [67]

2.3.3 Recently Proposed Deep Learning Architectures

There are several CNN architectures with different numbers of CONV, pooling, ReLU, and fully connected layers. The most common are:

- LeNet. In the 1990's the first successful application of CNNs were designed by Y. LeCun et al. [32]. The architecture was successfully applied to identify hand written check number and digits.

- AlexNet. Developed by Alex Krizhevsky, Ilya Sutskever and Geoff Hinton in 2012

[30]. AlexNet was submitted to the ImageNet Large Scale Visual Recognition Competition (ILSVRC) and achieved significant results. It is deeper and the convolutional layers are on top of each other while previous models used a single CONV layer followed immediately by a pool layer.

- ZF Net. The winner of ILSVRC 2013, designed by Matthew Zeiler and Rob Fergus[71]. It is an improved version of AlexNet by changing the size of the middle CONV layers and reducing the size of stride and filter on the first layer.

- GoogleNet. The winner of ILSVRC 2014, developed by Szegedy et al.[62] from Google. This architecture uses an Inception module to reduce the number of parameters. Besides, it uses a pooling layer instead of fully connected layers at the top of convolutional layers. The two next models are other versions using an Inception module. The idea of the inception module is that instead of choosing one convolution filter we can use all of them at the same time. In other words, it processes a specific input with multiple convolution filters, while it applies the pooling at the same time as well. At the end, the results of all these processes are concatenated. The advantage of this method is multi-level feature extraction from each input at the same time [62].

- VGGNet. The runner up of ILSVRC 2014, developed by Karen Simonyan and Andrew Zisserman [55]. This paper shows that the depth of the model is a very important factor in its performance. Their model consists of 16 CONV and fully connected layers. Even though this model has a high performance, it needs more memory and uses more parameters than other models.

- ResNet. Residual Network, winner of ILSVRC 2015 designed by Kaiming He et al. [20]. This model is different than other architectures in terms of skipping connections and over use of ReLU layers.

In this thesis, we are using the AlexNet model to train our images. We will describe this model in more detail in Chapter 5.

Chapter 3

Literature Review

There is much research that shows the worldwide growth of age-related neurodegenerative diseases like Alzheimer's Disease (AD)[70]. Developments and improvements in research are necessary for prevention, treatment, drug discovery, and health-care services. Interdisciplinary efforts in medicine, health care, computer science, economics and other disciplines have combined to address this rising crisis.

The brain has a complex composition. Thus, interpreting and understanding the brain is a challenge. However, for prevention, detection, treatment and behavioral understanding of neurodegenerative diseases, studying the brain has proven to be beneficial. To measure brain activity many techniques are used such as the electroencephalogram (EEG) test, Computed Tomography (CT), Cerebrospinal fluid (CSF), Magnetic Resonance Imaging (MRI), Magnetoencephalography (MEG), positron emission tomography (PET), and Functional MRI (fMRI) scan. These techniques play important roles in recognizing the risk factors and neural basis of disease, including Parkinsonian disease [15], Schizophrenia [66], Autism [9], and Alzheimer's disease [65] [45].

Machine learning classification algorithms have a significant role in analyzing brain imaging. Classification approaches have been used to extract features from neuroimaging data such as fMRI [47]. Choosing the right classifier is an important step in analyzing neuroimaging data. Different classifiers such as linear SVM, non-linear SVM, and artificial neural networks have been used in neuroimaging studies [47]. In this chapter we provide a review of many of these approaches and their application in classifying different brain images. In the last section we will review the studies that have been done using deep learning methods in neuroimaging data classification problems especially in Alzheimer's disease.

The fMRI is a non-invasive technique to measure brain activities. MRI creates a map of the brain anatomy while fMRI maps brain function over time. In other words, fMRI is

MRI structure over a time course. Thus, the dataset for an individual subject has a very large size [61].

As mentioned in the previous chapter, fMRI data creates blood-oxygen-level-dependent (BOLD) signals while associating neural activities in a specific brain area with some cognitive functions such as sensory, memory, and motion. Therefore, many different approaches have been applied to analyze these signals and the fMRI scans. In the following sections we describe previous works and different methods in the neuroimaging field.

3.1 Conventional Methods

One of the traditionally used methods in neuroimaging and analyzing BOLD signals is statistical: the General Linear Model (GLM). This model looks for linear correlations between a standard model and the fMRI time course to detect the activated areas in the brain [41]. GLM is a single variate conventional method which uses each voxel as a measurement. Thus, GLM needs one statistic for each voxel. In addition, the method uses numerous comparisons which increases the processing time [42].

Eckert *et al.* [15] studied the differential diagnosis of Parkinsonian disorders using PET imaging data. They used standard statistical parametric mapping (SPM) to analyze PET scans. For predicting different stages of disease (idiopathic Parkinson's disease (PD), multiple system atrophy (MSA), progressive supranuclear palsy (PSP), and corticobasal degeneration (CBGD)) they used GLM to compare the scans with the clinical information about the state of each individual. They obtained an accuracy of 85.4% to classify all subjects. The accuracy of each class in their study was 88.4% for early PD, 97.2% for late PD, 76% for MSA, 60% for PSP, 90.9% for CBGD, and 90.9% for normal control subjects.

Whalley et al. [66] used the same method as Eckert et al. [15] for image processing and analysis to predict Schizophrenia using fMRI data. They studied the fMRI scans of 21 normal control subjects, 21 high risk subjects with psychotic symptoms, and 41 high risk subjects without psychotic symptoms. As result of their experiment they found the p-value of control subjects versus high-risk without symptoms is 0.053, and control subjects versus high-risk with symptoms is $p = 0.118$.

The other statistical method that has been used in neuroimaging studies is Independent Component Analysis (ICA). The goal of the research Oghabian et al. [45] was to study the difference between Alzheimer's disease (AD), Mild Cognitive Impairment (MCI) and healthy control subjects using the ICA algorithm. They examined the fMRI scans of 15 normal control, 11 MCI, and 14 AD subjects. They applied seven stages of preprocessing including head motion correction using MCFLIRT [26], slice-timing cor-

rection, mean intensity normalization, spatial smoothing, brain extraction using FSL-BET [57], high-pass temporal filtering, and Gaussian low-pass temporal filtering. After the preprocessing steps, they compared fMRI activation patterns in resting-state of the three groups of subjects using the ICA algorithm. Their results showed that the significance of difference between normal and MCI class is 0.0097, between normal and AD is 0.0051, and between AD and MCI is 0.0168.

3.2 Multi-Voxel Pattern Analysis (MVPA) Methods

Multi-voxel pattern analysis (MVPA) is one of the most commonly used techniques in neuroimaging. MVPA uses information obtained from neural activities to find out the areas in the brain that are related to specific functional activities. MVPA usually uses a linear regression method as a classifier. Many research studies have used the multi-voxel pattern analysis in fMRI and neuroimaging data [14][8][13].

Federico De Martino *et al.* [14] employed a pipeline for classifying fMRI data. In their classification they used a multi variant pattern selection algorithm. Their algorithm used a support vector machine (SVM) recursively to remove unnecessary voxels and identify the instructive patterns in the fMRI data. They designed an experiment using four different sounds stimuli (girl, cat, tone, and guitar) during a fMRI session. After collecting the data of scans. They used the same preprocessing steps as Oghabi et al. [45]. Their results showed an accuracy of 66% for Girl vs. guitar, 58% for Girl vs. cat, 65% for Girl vs. tone, 61% for Guitar vs. cat, 65% Guitar vs. tone, and 67% Cat vs.tone. This was an improvement compared to previous univariate methods.

The goal of the study of Coutanche *et al.* [13] was to investigate the application of MVPA methods in predicting patient symptoms. They used the fMRI data of Autism Spectrum Disorder (ASD) patients and analyzed the data for the face processing task in these patients. In comparison with normal control individuals, ASD subjects show fusiform gyrus hypoactivation when they see unfamiliar faces. As a result of their study, they found a correlation between the MVPA classification method and the measures of symptoms. Their research showed that MVPA methods are able to classify patient severity using fMRI data.

In MVPA models, the challenge is to choose the best classifier for better performance. Therefore, new approaches of supervised learning have been introduced. Many powerful classifiers have been developed to improve the performance of the model. Naive Bayes is one of the best linear classifiers in this model. However, classifying fMRI data is not a linear problem.

New algorithms have been developed to be able to solve non-linear problems such as support vector machines (SVM) [12][31]. This method is an alternative to the Naive Bayes method with more parameters to be able to solve non-linear problems[31]. MVPA is a supervised classification model. In a supervised classifier, the purpose is to learn the relationships between neuron activities in voxels of fMRI and the functional activity.

In the following section we will review the application of deep learning as feature extraction in classifying fMRI and PET data using a SVM classifier. Also we will discuss previous works on deep learning methods and convolutional neural networks in the neuroimaging field.

3.3 Deep learning

Recently, with increased use of GPUs, deep learning approaches have been significantly applied in neuroimaging fields and to analyze big data. As mentioned in the previous chapter, in deep learning methods the feature extraction is an automatic process. Therefore, the models perform more quickly and more accurately. In [48] Sergey M. Plis *et al.* show the success of the application of deep learning approaches in the neuroimaging field both in structural and functional brain imaging data. First, they used shallow restricted Boltzmann machine (RBM) and compared it with the performance of models from single matrix factorization class including ICA, Principal component analysis, sparse PCA, and sparse Non-negative matrix factorization. Data used in their work was task-related scans of 28 healthy participants who were scanned during an auditory oddball task (AOD). Their study showed that RBM and ICA had the best overall performance. Then to study the depth effect they used deep belief networks with restricted Boltzmann machine as a top level in their study. They applied their methods in two different case studies. In the first case study, they evaluated the effect of depth in the DBN method to classify structural MRI (sMRI) data from a schizophrenia dataset. The dataset consisted of 198 schizophrenia patients and 191 healthy controls. They validated the accuracy of their classification with 10 fold cross validation. In order to study the effect of depth they trained their data with the RBF-kernel SVM, logistic regression (LR) and a k-nearest neighbors (KNN) classifier. The 10-fold cross validation was performed both with raw data and in 3 depth. The F-score of classification using SVM on raw data was 0.68 and after one layer of depth was 0.66, after second layer 0.62, and third layer 0.90. As you can see there was significant improvement after the second layer. For logistic regression the raw data result was 0.63 and the result of first, second, and third layers were 0.65, 0.61, and 0.91 respectively. The last experiment with KNN for raw data resulted in 0.61

F-score and 0.55, 0.58, and 0.90 after the first, second, and third layers respectively.

In their second case study, sMRI data was collected from healthy controls (859 scans) and Huntington's disease (2641 scans) datasets. They showed that the accuracy of their models increased by increasing the depth of the model. They investigate the effect of depth in this case study by comparing F-score of SVM and logistic regression of raw data and classification of data after the first, second and third layer. Their results for SVM showed that with raw data the F-score was 0.75 while using LR increased the F-score to 0.79. After the first and second layers for both SVM and LR, they obtained on F-score of 0.65. However, after the third layer the F-score of both experiments increased to 1.00.

In their study they showed that deep learning methods learn the main physiologically representations and extract and detect the hidden relations in neuroimaging data.

3.3.1 Deep learning to classify Alzheimer's disease

As Alzheimer's disease is one of the main causes of death in the United States [2] researchers have become interested in understanding its development, and in the prediction of the disease by monitoring the brain changes during its progression. In computer science studies, researchers have become interested in studying the classification and prediction of the disease.

The researchers in the studies described in [59] [60] classified Alzheimer's disease (AD), mild cognitive impair (MCI), and MCI converter (those individuals whose stage of disease changed to AD after 15 months) classes using both MRI and PET datasets. They used the ADNI dataset (which will be discussed in Chapter 4) for both MRI and PET imaging. Their data was acquired from 93 AD subjects, 76 MCI converters (MCI-C) and 128 MCI non-converters (MCI-NC), and 101 NC subjects. Before applying the classification method the MRI data was preprocessed by the typical procedures of skull-stripping, and cerebellum removal, and spatial normalization to standard data. They used MIPAV software and FAST in FSL package for preprocessing the data. They designed an auto-encoder network for extracting features from images. The SVM classification was applied as a learning method. Their method achieved accuracies of 95.35% (AD vs. NC), 85.67% (MCI vs. NC), and 75.92% (MCI-C vs. MCI-NC).

In 2015, Siqi Liu *et al.* [37] implemented a new multi modal feature extraction for multi class Alzheimer's disease (AD) diagnosis. In their model, in order to save all the information and features in the data a zero-masking strategy was applied. To extract higher-level features they used stacked auto-encoder (SAE) networks. The SVM classification and multi-modal feature extraction algorithms were applied. The best result of

their study was 86.86% accuracy.

Despite the fact that use of SVM classifiers in fMRI problems improved the accuracy of the classifier it was not always a better solution. The possible reason is that fMRI data is considered as big data. Also, SVMs need a feature selection step which is time consuming [48].

Payan *et al.* [46] developed an algorithm for classification of Alzheimer's disease into three classes: Alzheimer's disease(AD), mild cognitive impair (MCI), and normal control (NC) subjects. They applied a 3D convolutional neural network with an auto-encoder and 2D CNNs on their datasets. They achieved an accuracy of 89.47% for the 3D classification model. Their results for the 2D CNN model were the same as the results that Siqi Liu *et al.* [38] obtained from their 2D experiments which was approximately 85.53% accuracy.

Another recent study on classifying Alzheimer's disease was performed by Sarraf *et al.* [51] [52]. They designed a pipeline to classify both fMRI and MRI data of Alzheimer's disease subjects and normal control subjects. They applied two different methods for their binary classifiers. The CNN architectures that were applied included LeNet-5 and GoogleNet. They achieved outstanding results using both models. An average accuracy of 99% and 100% were acquired respectively in the LeNet and GoogleNet models for fMRI data.

Chapter 4

Datasets

In this chapter, first we describe the dataset acquisition for our experiments. The second section of this chapter provides a pipeline for preprocessing the raw data acquired in the first section.

4.1 Data Acquisition

In this study, a subset of the Alzheimer's Disease Neuroimaging Initiative (ADNI) database [1] was used to train and validate our convolutional neural network (CNN) classifier. This subset included resting-state functional magnetic resonance imaging (rs-fMRI) scans of 197 subjects with average age > 74. Based on their mini mental state examination (MMSE), these subjects were in different stages of disease. As each subject might visit multiple times and by considering each visit as a separate subject we could have 355 subjects to study. However, upon further investigation we found that the stages of disease in different visits generally did not change since the dates of the visits were close to one another or even on the same date. Thus we removed the repeated and corrupted (during preprocessing) data and proceeded to use our model with 197 subjects. As stated in [1] "The purpose of the ADNI study is to track the progression of the disease using biomarkers to assess the brain's structure and function over the course of four disease states". As mentioned in Chapter 2, Table 2.1, ADNI categorize different stages of Alzheimer's disease for participants in healthy normal control (NC), significant memory concern (SMC), early mild cognitive impairment (EMCI), late mild cognitive impairment (LMCI), and Alzheimer's disease (AD) classes.

The scans in ADNI were performed on two different Tesla scanners, namely Philips Medical systems and SIEMENS. Philips Medical system scans were obtained with an EPI sequence of 144 volumes, Field Strength=3.0 tesla, Flip Angle=80.0 degree, TE=30.0 ms,

TR=3000.0 ms, 64 × 65 matrix, and 6720.0 slices of 3.31 mm thickness for Resting State fMRI. The EPI sequence of Extended Resting State fMRI with the Philips Medical system scanner was 200 volumes, Field Strength=3.0 tesla, Flip Angle=90.0 degree, TE=30.0 ms, TR=3000.0, 64 × 65 matrix, and 9600.0 slices of 3.31 mm thickness. For SIEMENS scanner the EPI sequence was 197 volumes, Field Strength=3.0 tesla, Flip Angle=80.0 degree, TE=30.0 ms, TR=2999.99, 448 × 448 matrix, and 197 slices of 3.4 mm thickness. We considered these differences during the preprocessing steps to avoid any influence on our results.

In this study we chose 197 subjects from NC, SMC, EMCI, LMCI, and AD participants. 107 of the participants were female and 90 were male with an average age of 74.4. We eliminated the MCI class since there were not enough subjects available (only two subjects with MCI). The fMRI data were available for ADNI GO and ADNI 2 projects. However, the only project with MCI individuals was ADNI 1. Table 4.1 shows the number of subjects in each category that we used in this study.

Group	Subj.	Female	Mean Age	Male	Mean Age
NC	55	30	74.73	25	78.52
SMC	25	17	75.06	8	75.45
EMCI	46	29	71.66	17	74.24
LMCI	39	14	74.36	25	75.04
AD	29	16	72.16	13	75.45

Table 4.1: Number of subjects and their mean age that we used in this study.

We obtained our dataset with the above description in the Digital Imaging and Communications in Medicine (DICOM) fMRI format. In order to analyze the data we needed to convert them to Neuroimaging Informatics Technology Initiative (NIFTI) format in order to work with the available neuroimaging toolbox, and then apply some preprocessing on the raw data. This will be described in the following section.

4.2 Data preprocessing

In order to take raw data from the scanner and prepare it for analysis, preprocessing is the most important step. Preprocessing is a series of data transformations to reduce

the noise in the raw data. Preprocessing increases the sensitivity of analysis (SNR) and certifies the validity of the statistical model. As previously mentioned, the raw fMRI data is in DICOM format. However, the required format in most fMRI analysis tools is the NIFTI format. We first converted the DICOM data to NII format by using the dcm2nii toolbox developed by Chris Roden et al. [10]. Figure 4.1 is a sample of a raw fMRI for an Alzheimer's disease subject before any preprocessing step is applied.

4.2.1 Brain extraction

Since we are studying the brain tissue, we need to remove the non-brain areas such as skull and neck voxels from fMRI data. The FSL-BET toolbox [57] was used to extract the brain area from fMRI images as shown in Figure 4.2.

4.2.2 Motion correction

The next step in preprocessing is motion correction. It is common for subjects to move their heads during a fMRI session. Within the functional images over time, the motion changes the position of the brain. Therefore, over time a voxel's time series may not refer to the same point of the brain. Despite the scale of the motion (sub-voxel motion or obvious motion), this can have destructive effects on the statistical analysis. Thus, motion correction is a very important step in fMRI data preprocessing. In this part of the study, we used the FSL-MCFLIRT toolbox [26]. The FSL-MCFLIRT tool applies rigid-body transformation for motion correction.

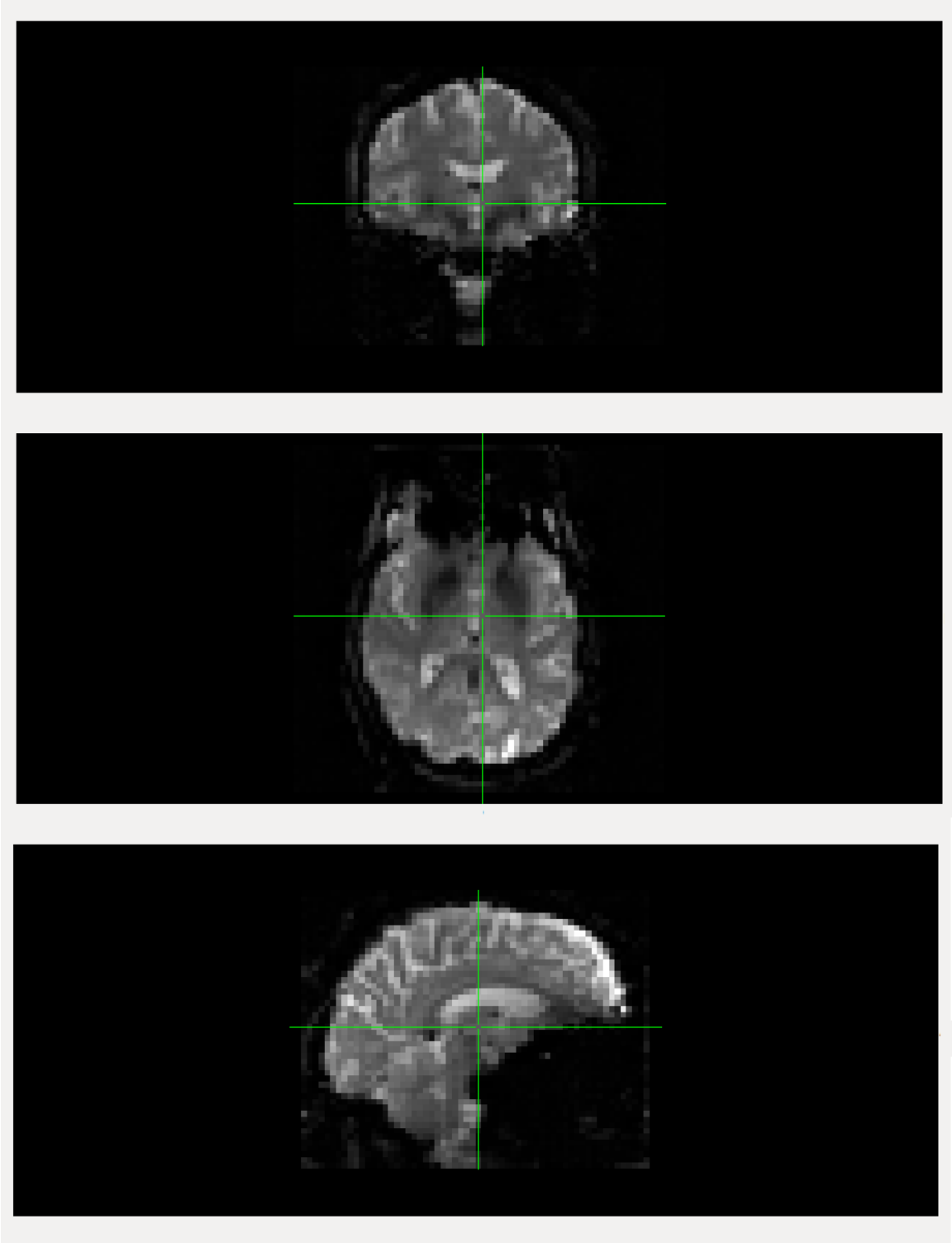


Figure 4.1: Example of raw fMRI scan of a patient with Alzheimer's disease (subject number 002-S-5018) in different cronal (top), axial (middle), and sagittal(bottom) view using FSLView toolbox

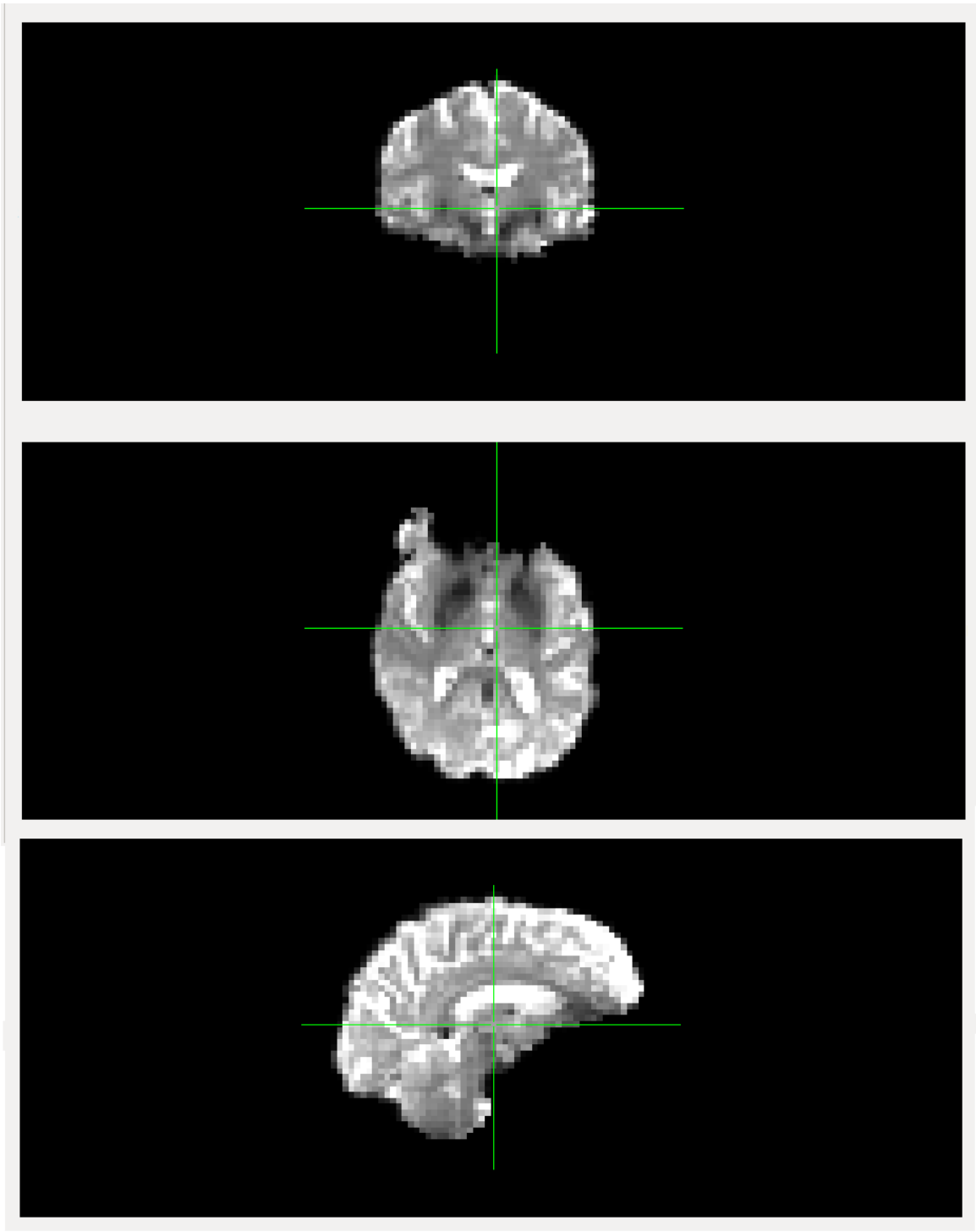


Figure 4.2: Example of fMRI scan of a patient with Alzheimer's disease (subject number 002-S-5018) after extracting the brain area from raw data using the FSL-BET toolbox. Scan is in different cronal (top), axial (middle), and sagittal(bottom) view, using FSLView toolbox

4.2.3 Slice timing corrections

The fMRI volumes are scanned one slice at a time. The timing of this acquisition is equally spread over the repetition time (TR= 3000 ms). Therefore, each voxel is scanned at a different time. The changes in the timing should be correctly justified to match the stimulus and response timing so the statistical analysis will be able to fit the model more accurately. The purpose of the slice timing correction is to justify the voxel time series to have a reference timing for all voxels. Slice timing corrections use some form of interpolation to shift the time series of values either forward or backward in time to achieve the temporal adjustment. In this study we used Hanning- windowed Sinc interpolation [33].

4.2.4 Spatial smoothing

At this point in preprocessing, the spatial smoothing of each volume is performed. The aim of performing spatial smoothing is to reduce the noise level while protecting the underlying signal. In order to protect the underlying signal from being reduced along with the noise, the extent of the spatial smoothing should not be larger than the size of the activated region. The most common way of smoothing is deconvoluting 3D images with a 3D Gaussian filter. The degree of smoothing is proportional to the full width at half-maximun (FWHM) of the Gaussian distribution. In this work, we performed spatial smoothing by using a Gaussian kernel of 5mm FWHM.

4.2.5 High-Pass filtering

To remove low level unwanted signals in the voxels' time series we applied high pass filtering. These low level noises could be the results of some physiological artefacts such as breathing, heartbeats, or by physical noises like scanner drifts. It is necessary to remove these low level noises since they will appear in the statistical analysis later and affect the fitting of the data to the model. The signal drifts have a low frequency. Therefore, we can remove them by applying a high-pass filter. This filter allows the high frequencies (stimulus activities) to pass, but removes low frequencies (the signal drifts). In this work, we used temporal high-pass filter with 0.01 HZ cut-off frequency.

4.2.6 Spatial normalization

The next step in fMRI preprocessing is spatial normalization. During the fMRI session, we have a particular high-resolution structural scan (T1-weighting (T1w)) according to

the subject's brain shape and layout. However, to analyze the data of many subjects we need to have the same conditions. Only with equivalent brains (common spatial domain) are we able to compare changes in specific voxels in particular areas for all subjects. Registration of our anatomical brain to a template as a process in the fsl FLIRT toolbox [57][25] can perform the alignment of fMRI scans to a common spatial domain (reference template).

For spatial normalization first we need to put our fMRI data in T1w space by using FSL's FLIRT [57][25]. We used a linear transformation with 7 degrees of freedom (7 DOF). Then we registered MNI152 standard space (derived from average of 152 structural images after high-dimensional nonlinear registration) as our template shown in Figure 4.3. The MNI152 is a common template provided in FSL toolbox which has been used in previous works [50] [52]. To register the template we performed a linear transformation with a wider degree of freedom (12 DOF) rather than the reregistration that we applied to create our own T1w space. This helps to fit our data all over the shape and the size by applying translations, rotations, zooms, and sheers.

As a result of these preprocessing steps, as shown in Figure 4.4 we obtained a 4D NIFTI file for each subject in which each of these NIFTI files contains $64 \times 45 \times 64$, 3D volume per time course.

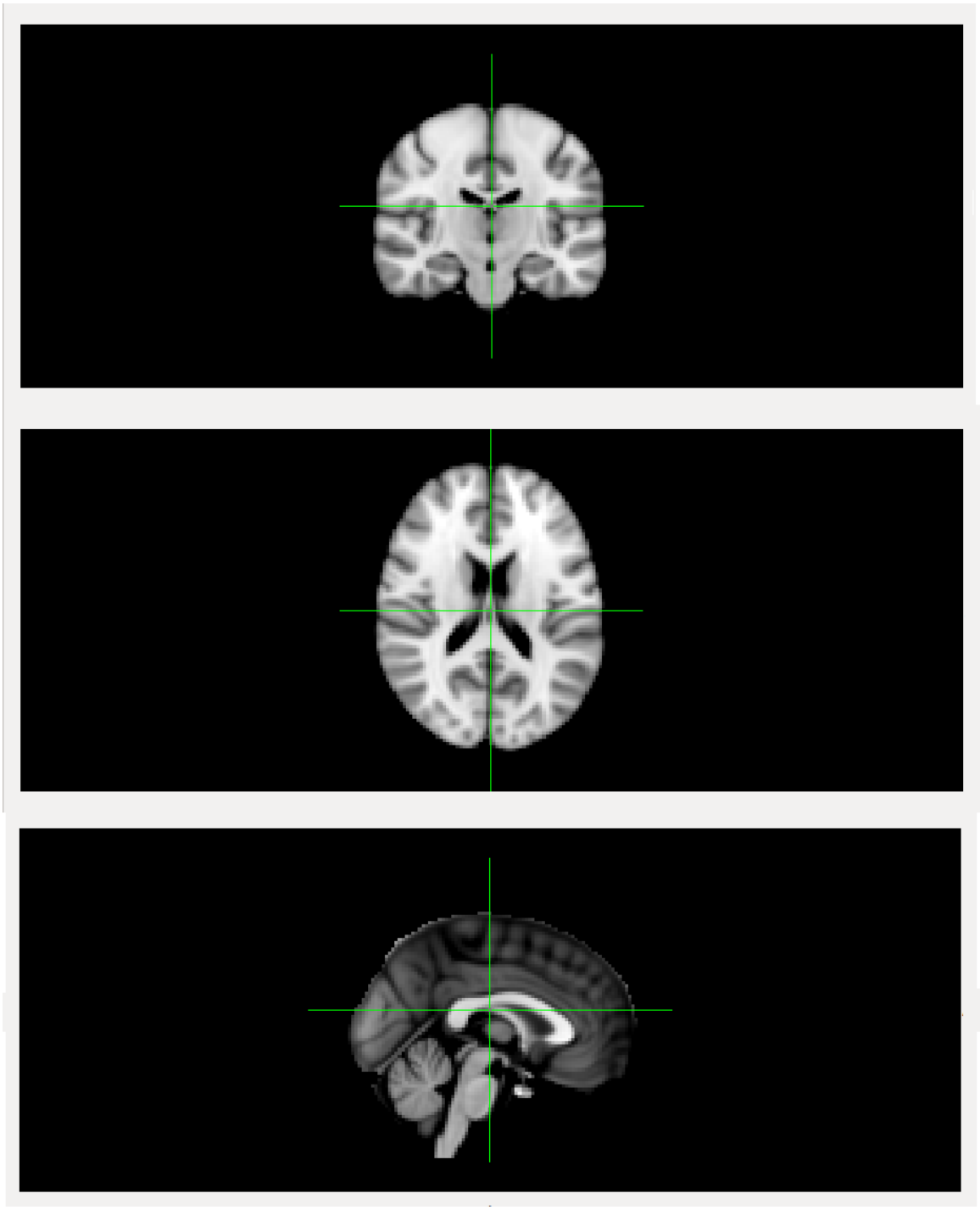


Figure 4.3: Average of 152 structural images after high-dimensional nonlinear registration which is known as MNI152 standard space.

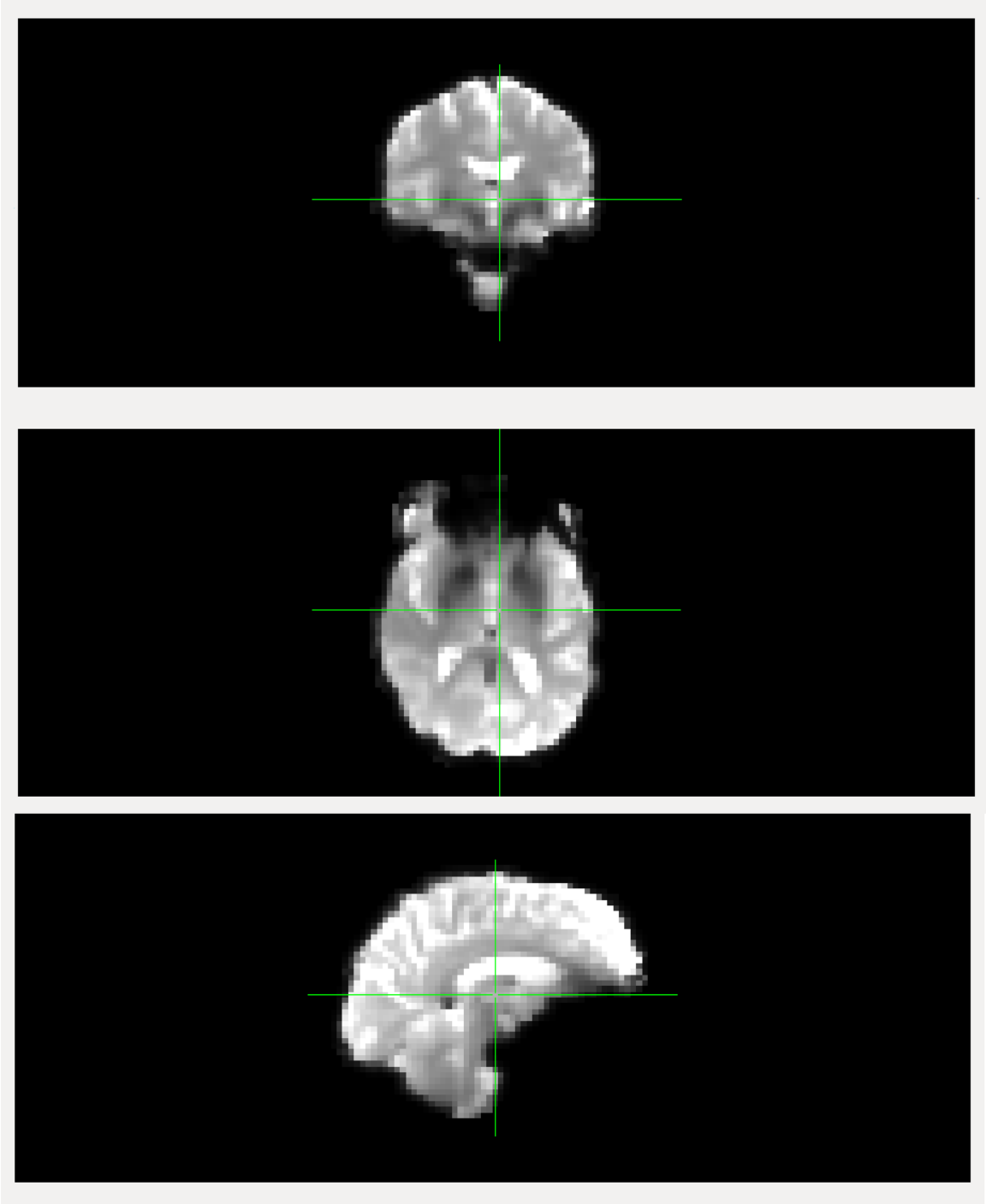


Figure 4.4: Example of an Alzheimer's disease (subject number 002-S-5018) scan after all the preprocessing steps.

4.2.7 Image Conversion

In order to create a proper dataset for the experiment we decomposed 4D NIFTI files into 2D matrices along z axis and time course. Next, 2D matrices were converted to lossless Portable Network Graphics (PNG) format using MATLAB's NIfTI and analyze image toolbox [28]. The last 3 slices of each time course for each subject were removed as they did not carry any functional information. The result of this conversion for each subject was 43 slices of 64×64 PNG images per time course. For example the resting state fMRI data acquired by Philips system with 140 volumes (time course), $43 \times 140 = 6020$ images of size 64×64 was obtained for each individual. Therefore, by converting NIFTI data for all subjects, 1,322,085 PNG images of size 64×64 were collected.

In Chapter 5 we will describe how we divided the dataset into training, validation, and classification subsets.

Chapter 5

Methodologies

The first part of this chapter describes the preliminary studies on two deep learning models, AlexNet and GoogleNet and explains why we choose AlexNet as the model for our experiment. The second part is an introduction to the AlexNet model and its different layers. The third part explains the experiment that we designed for classifying our data into five different stages of Alzheimer's disease.

5.1 Preliminary Evaluation

Based on the literature review conducted in Chapter 3, we are using deep learning as a feature extractor to improve the results of the classification of the different stages of Alzheimer's disease. Previous research classified two classes, normal control and Alzheimer's disease, using LeNet-5 and GoogleNet methods. The results were significantly improved compared to other methods.

Therefore, we decided to do some preliminary studies on these new methods and their application to medical imaging classification problems. We considered two different models, AlexNet and GoogleNet. These were chosen because they have been successful on similar problems [51] [52] [54]. The multi-layers, automatic feature extraction, optimization of the extracted features, and normalization layers after each convolutional layer improved the classification especially when there was low variance between classes.

The accuracy of AlexNet for our first dataset was 97.70% and for GoogleNet was 97.73%. In order to compare the accuracy of these two models per class, 20000 random images of the unseen test dataset were used in both trained models. Then the confusion matrices and the accuracy per class of the test dataset were calculated for both models (Tables 5.1 and 5.2).

	AD	EMCI	LMCI	NC	SMC	Accuracy per Class
AD	2635	67	52	29	14	94.21%
EMCI	69	4103	42	8	13	96.88%
LMCI	87	88	4231	14	10	95.51%
NC	153	144	62	5302	24	93.26%
SMC	82	87	30	24	2630	92.18%

Table 5.1: The confusion matrix for testing 20,000 images in GoogleNet trained model.

	AD	EMCI	LMCI	NC	SMC	Accuracy per Class
AD	2651	11	25	99	11	94.78%
EMCI	11	4061	19	133	11	95.89%
LMCI	0	15	4253	156	6	96.0%
NC	22	27	40	5591	5	98.35%
SMC	23	27	13	92	2698	94.57%

Table 5.2: The confusion matrix for testing 20,000 images in AlexNet trained model.

Both AlexNet and GoogleNet had good performance in classification of our preprocessed data. However, GoogleNet was more time consuming. Therefore, for this research we choose AlexNet as our classifier.

Other new deep learning models and fine tuning pre-trained models will be further discussed in the future work section of Chapter 7.

5.2 Deep Learning Architecture (AlexNet)

AlexNet was designed and developed by Krizhevsky *et al.* [30] in 2012 for the ImageNet Large Scale Visual Recognition Competition (ILSVRC). AlexNet is a deep convolutional neural network (CNN) which was used to classify the 1.2 million images into 1000 classes in the ImageNet challenge [30]. AlexNet was able to achieve considerably better results than previous models such as LeNet [32]. The differences between AlexNet and previous models is the number layers and parameters. AlexNet consists of five CONV layers

of which some of are followed by max-pooling layers, three fully connected layers, and a final 1000-way softmax [30]. AlexNet's total trainable parameters number about 60 million. In comparison, for example LeNet consists of two convolutional layers followed by two pooling layers and three fully connected layers. LeNet's total trainable parameters number about 60,000. AlexNet uses non-linear function (ReLU), while LeNet uses logistic sigmoid function. The last difference is that AlexNet uses a regularization method called "Dropout" to reduce the overfitting in the fully connected layers, while this concept was not used in LeNet architecture.

Before going through an explanation of AlexNet's topology we need to describe some more concepts used in the model. The following subsections describe the AlexNet features in more detail.

5.2.1 Local Response Normalization

Even though the ReLU activation function does not need input normalization to avoid saturating zone, AlexNet topology uses a local normalization after applying ReLU in CONV layers. The response normalized activity $b^i_{x,y}$ is calculated by the following equation [30]:

$$b^i_{x,y} = \frac{a^i_{x,y}}{(k+\alpha\sum_{j=max(0,i-n/2)}^{min(N-1,i+n/2)}(a^j_{x,y})^2)^\beta} \tag{5.1}$$

In this equation $a^i_{(x,y)}$ is the activity of neurons which is calculated by applying filter i at (x,y) and N is the total number of features in the layer. The values of the constant hyper parameters k, n, α, and β are determined by using a validation test. Krizhevsky et al. used $k = 2, n = 5, \alpha = 10^{-4}$, and $\beta = 0.75$ in their experiment[30]. In Equation 5.1 the summation runs over n adjacent feature maps in the same position.

5.2.2 Reducing Overfitting in AlexNet

Overfitting means that the model learns the entire image during training instead of learning features in the image. This happens when the number of weights and parameters in the CNN model is high while the number of samples is not high enough. Overfitting reduces the model's performance during testing the dataset. The number of parameters in AlexNet is 60 million. To avoid overfitting in such models the number of samples needs to be 10 times more than the number of parameters [5].

Therefore, there are many techniques to reduce the overfitting in CNNs. The first one

is increasing the dataset or reducing the complexity of the model, which is not possible to do all the time. The second possible way is using data augmentation. AlexNet uses two different forms of augmentation. The first form generates image translation and horizontal reflection[30]. The second form changes the RGB intensities in the training images.

The last method to reduce the overfitting is adding regularization. AlexNet adds the dropout method [23] as regularization to the model. Dropout sets the output of each hidden neuron to zero with some specific probability (0.5 in AlexNet). This mean some neurons will be dropped out and they will not present during forward pass and back-propagation. In other words, the model trains a different architecture for every input. However, all these architectures share weights. Dropout aids in reducing complex co-adaptations by removing the dependency of a neuron on the presence of the other neurons. This technique makes the neuron generate more features.

5.2.3 Overall Architecture

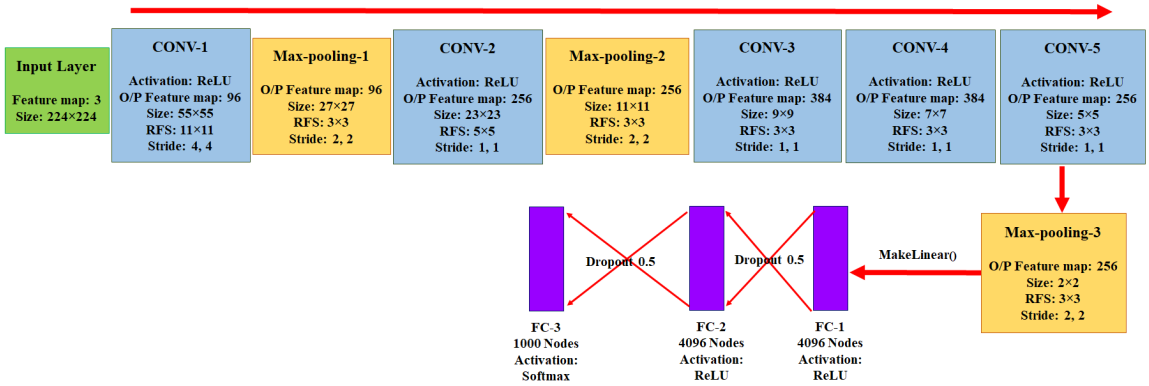


Figure 5.1: An illustration of the AlexNet architecture. Amended from [56].

Figure 5.1 shows the overall architecture and configuration of the AlexNet layers. As can be seen a subset of generated activation maps (feature maps) in each layer is sent to the next layer as input. Figure 5.1 shows that the activation function used after each CONV layer and fully connected layer is rectified linear unit (ReLU) which are described in Section 2.3.1. The response normalization layers apply to the first and second CONV layers. The max pooling layers follow after each response normalization layer and the fifth CONV layer. Also, the most important feature in this model is reducing the overfitting by applying the dropout regularization method after fully connected layers. This configuration reduces the number of input features from $224 \times 224 \times 3$ to $2 \times 2 \times 256$ after

the third max pooling layer.

The first CONV layer takes an input image with a size of $224 \times 224 \times 3$ and filters the input with 96 feature maps of Receptive Field Size (RFS) of $11 \times 11 \times 3$ with a stride size of 4 pixels (distance between the centers of receptive fields of neurons in a feature map). The output of the first CONV layer (input of second layer) is calculated as $\frac{224}{4} \times \frac{224}{4} \times 96 = 55 \times 55 \times 96$. The output of the first layer after the max pooling layer and response normalization layer is sent to the second CONV layer. The second CONV layer filters the input with 256 feature maps of size $5 \times 5 \times 48$. The last three CONV layers are connected to each other without any pooling or normalization layers in between. After the fifth CONV layer the output is passed to the last max pooling layer. The result of the max pooling layer is passed to the fully connected layers which each of has 4096 neurons.

5.3 Our Experiment

Once the data had been converted, as described in Section 4.2.7, the images were randomly shuffled. To validate the classification, five-fold cross validation against the whole dataset was applied and five subsets were randomly created. In each experiment, as shown in Table 5.3, 60% of the data was used for training, 20% for validation and 20% for testing.

	Training	Validation	Test
Percentage	60%	20%	20%
Images	793251	264417	264417

Table 5.3: Percentage and size of datasets for training, validation, and testing.

	Normal	SMC	EMCI	LMCI	AD
Training	227,016	113,265	169,848	171,882	111,240
Validation	75,672	37,755	56,616	57,294	37,080
Testing	75,672	37,755	56,616	57,294	37,080

Table 5.4: Number of images in each category for training, validation, and testing.

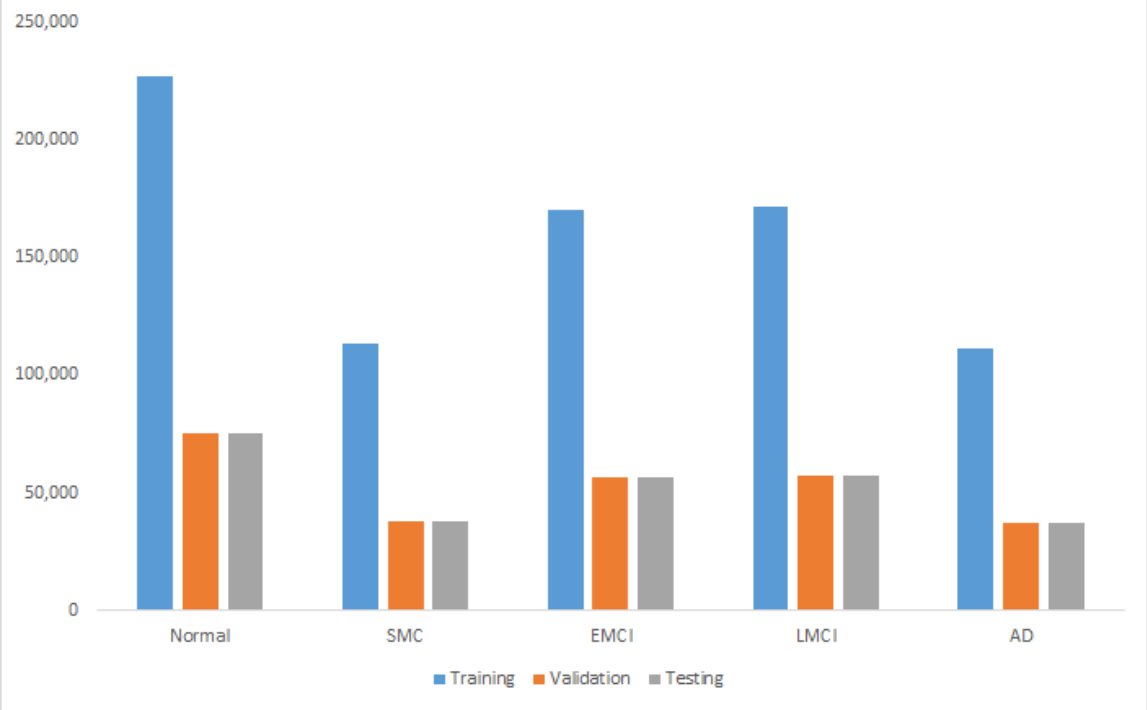


Figure 5.2: Illustration of training, validation, and testing datasets

Table 5.4 and Figure 5.2 show the number of images in each of the categories for our datasets.

As described in the previous section, the AlexNet Architecture was used as a CNN classifier in this study. We labeled our datasets into five classes (AD as 0, EMCI as 1, LMCI as 2, NC as 3, and SMC as 4).

In this study, we used the Caffe Deep Learning framework [27] to train our model. In order to use Caffe, we converted our datasets to Lightning Memory Mapped Database (LMDB) and resized images to 256 × 256 pixels. Then, we generated the mean image of the training data and subtracted it from each input image, so we can make sure that the mean of every feature pixel is zero. Figure 5.3 is the resulting mean image for one of the datasets of our first experiment.

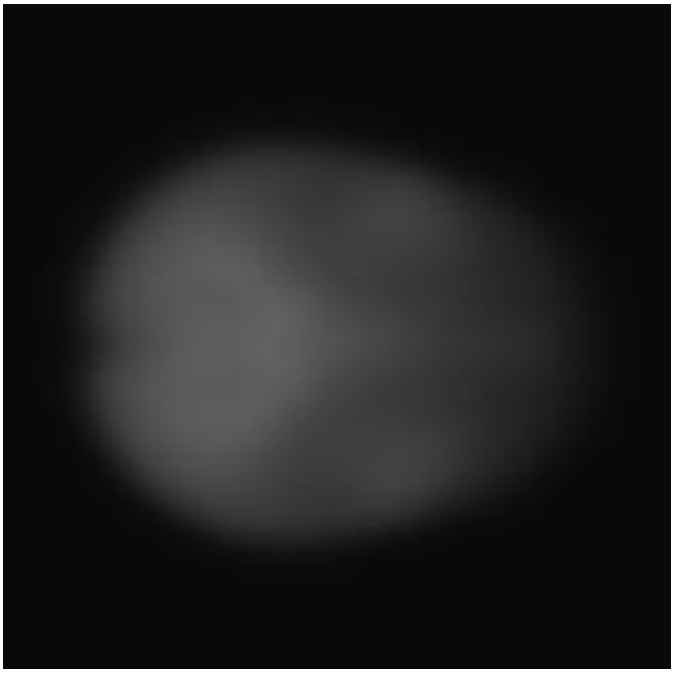


Figure 5.3: The mean image calculated from all training images in one dataset

5.3.1 Model and solver definition

In this subsection we explain how we implemented our model in the Caffe framework. We adjusted the AlexNet model for 30 epochs since the learning rate with our configuration (which will be described later) after epoch 21 approaches zero. Then we used the Caffe Stochastic Gradient Descent (SGD) solver method. The solver generates the training and testing networks for, respectively, learning and evaluation purposes. It takes snapshots of the model and the solver states during optimization. In order to learn the model, during each iteration, the solver computes the output and loss by calling the network forwards, and computes the gradients by calling the network backwards. Based on the solver method it will update parameters differently. Once the parameters are updated, the solver will update its state based on the learning rate, method, and history.

The SGD solver method uses Equation 5.2 to update the weights W.

$$\begin{aligned} V_t + 1 &= \mu V_t - \alpha \bigtriangledown L(W_t) \\ W_t + 1 &= W_t + V_t + 1 \end{aligned} \tag{5.2}$$

where $\alpha \bigtriangledown L(W)$ is the negative gradient, V_t is the previous weight update, α is the learning rate, and μ is the momentum as hyper parameters. This is as Krizhevsky et al.[30]

described in their model, for deep learning using SGD as the solver. The best value to initialize the learning rate α is close to 0.01, and then dropping it by some constant value after each iterations. The common learning rate policy is a step which drops the learning rate by a factor of Gamma in each stepwise iteration. Usually in deep learning, the learning rate and momentum parameter are related. In this study, the momentum parameter is $\mu = 0.9$. The momentum smoothes the weight updates during the iterations, which makes the model more stable and fast. The momentum after many iterations multiplies the size of weight updates by a factor of $\frac{1}{1-\mu}$. Then, it is important to decrease the size of α in the case of increasing the momentum.

In this study, we initialized the hyper parameters for AlexNet model in Caffe as $Gamma = 0.1$, $momentum = 0.9$, $Weight - decay = 0.001$, and $Learningrate = 0.01$. Figure 5.4 shows the policy applied for dropping the learning rate every stepsize iterations.

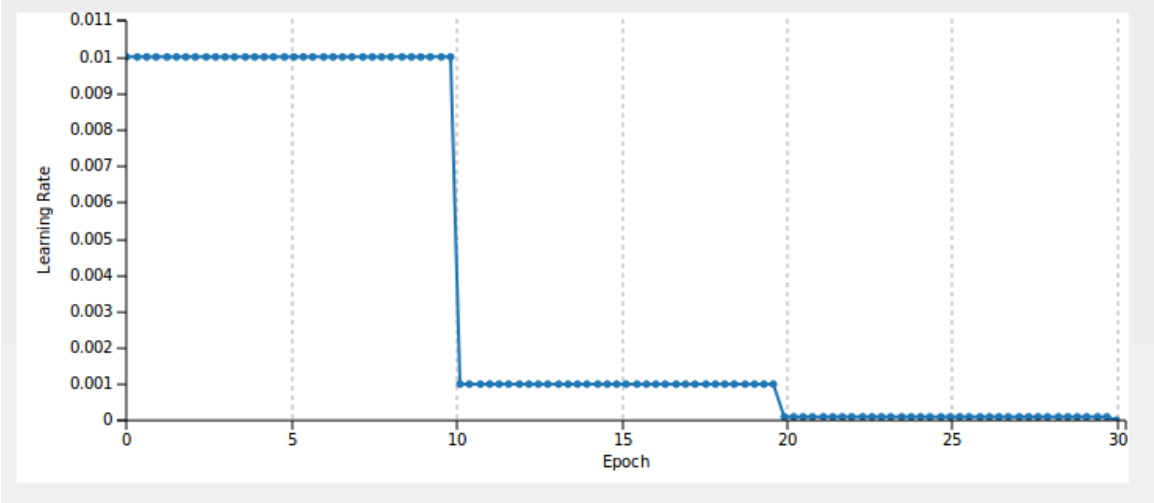


Figure 5.4: Learning rate is dropping by the factor of gamma

Figure 5.5 is a snapshot of the solver.prototxt file in Caffe, to show all the parameter initialization we used in our model.

```
test_iter: 8264
test_interval: 6198
base_lr: 0.01           # begin training at a learning rate of 0.01 = 1e-2
display: 40
max_iter: 185940        # train for 185940 iterations total
lr_policy: "step"       # learning rate policy: drop the learning rate in "steps"
                        # by a factor of gamma every stepsize iterations
gamma: 0.1              # drop the learning rate by a factor of 10
                        # (i.e., multiply it by a factor of gamma = 0.1)
momentum: 0.9
weight_decay: 0.0001
stepsize: 61361         # drop the learning rate every 61361 iterations
snapshot: 6198
snapshot_prefix: "snapshot"
solver_mode: GPU
net: "train_val.prototxt"
solver_type: SGD
```

Figure 5.5: Snapshot of solver.prototxt file

The architecture of our model after initializing the parameters is shown in Figure 5.6.

We repeated our experiment five times on the Amazon Web Service (AWS). The configuration of AWS was Linux G2.8xlarge, with four NVIDIA GPUs, each GPU has 4G memory, 1,536 CUDA cores, and 32 Intel Xeon E5-2670 vCPUs. Each experiment was trained on our server for about five to six days.

The results of these experiment will be discussed in Chapter 6.

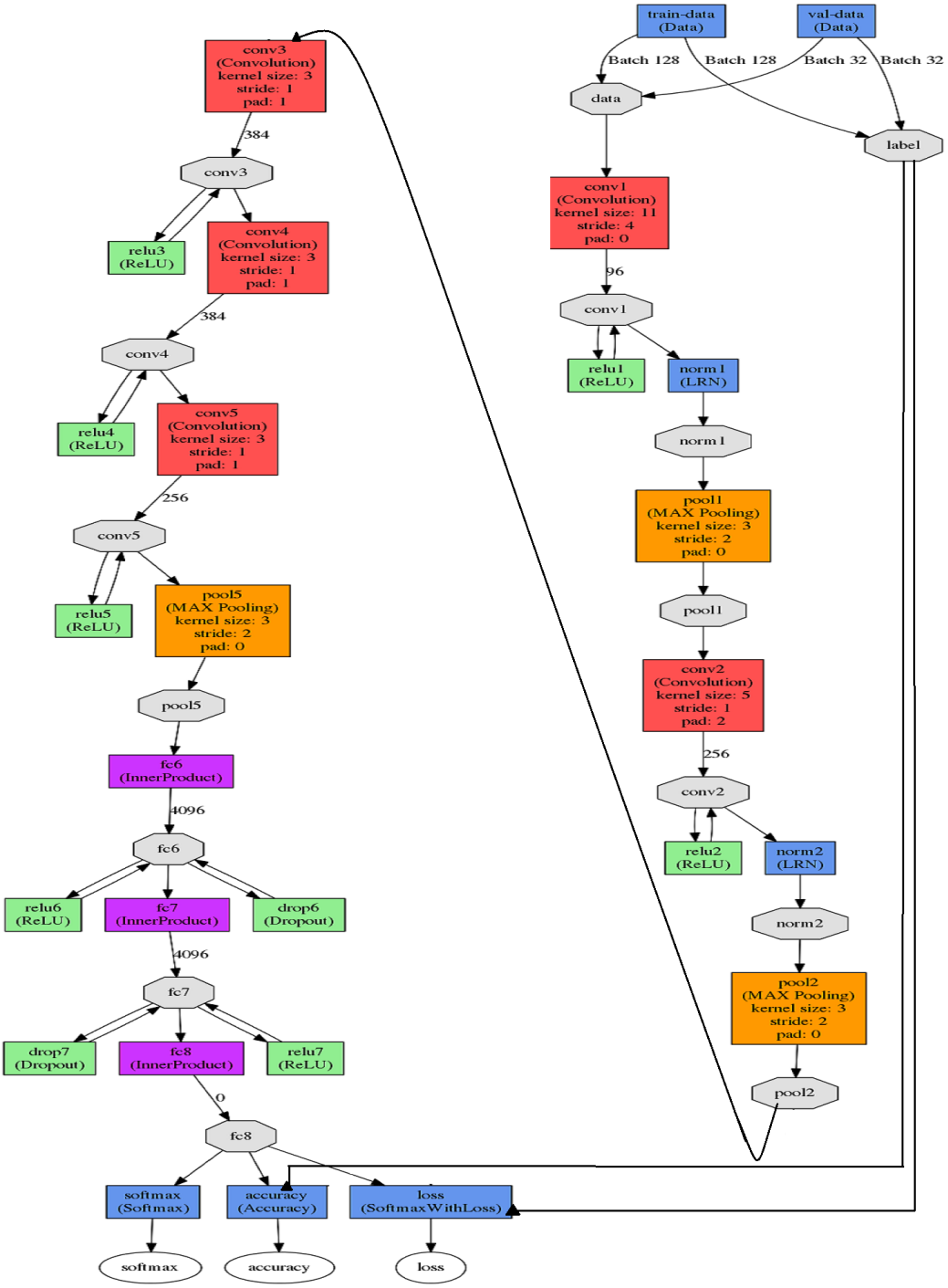


Figure 5.6: AlexNet model for training fMRI datasets

Chapter 6

Results and Discussion

After running our experiments for training on the Amazon Web Service (AWS) an average accuracy of 97.64% was achieved. Table 6.1 shows the results of the accuracy of our model in each experiment. As shown, we obtained a very good accuracy in all experiments.

Besides the accuracy of model, we also monitored the loss in the training and testing datasets during the learning process, as shown in Figure 6.1 to Figure 6.5.

Classifying different stages of Alzheimer's disease needs accurate preprocessing and careful feature learning due to the similarity in brain anatomy in older adults and also similarity intensities in the images. As we described in previous chapters, we used a very careful preprocessing on our dataset in order to prevent any problem during image analysis. The preprocessing of fMRI data has a major influence on the high and consistent accuracy.

Experiments	1	2	3	4	5	Average
Accuracy	97.7013	97.6215	97.6082	97.6317	97.6169	97.63592

Table 6.1: The accuracy of our model in each experiment and the average accuracy of all five experiments.

During classification, the AlexNet model was trained and tested with a large amount of data that we acquired from 4D fMRI images. A set of learnable filters was applied in the model to extract low to high level features from images.

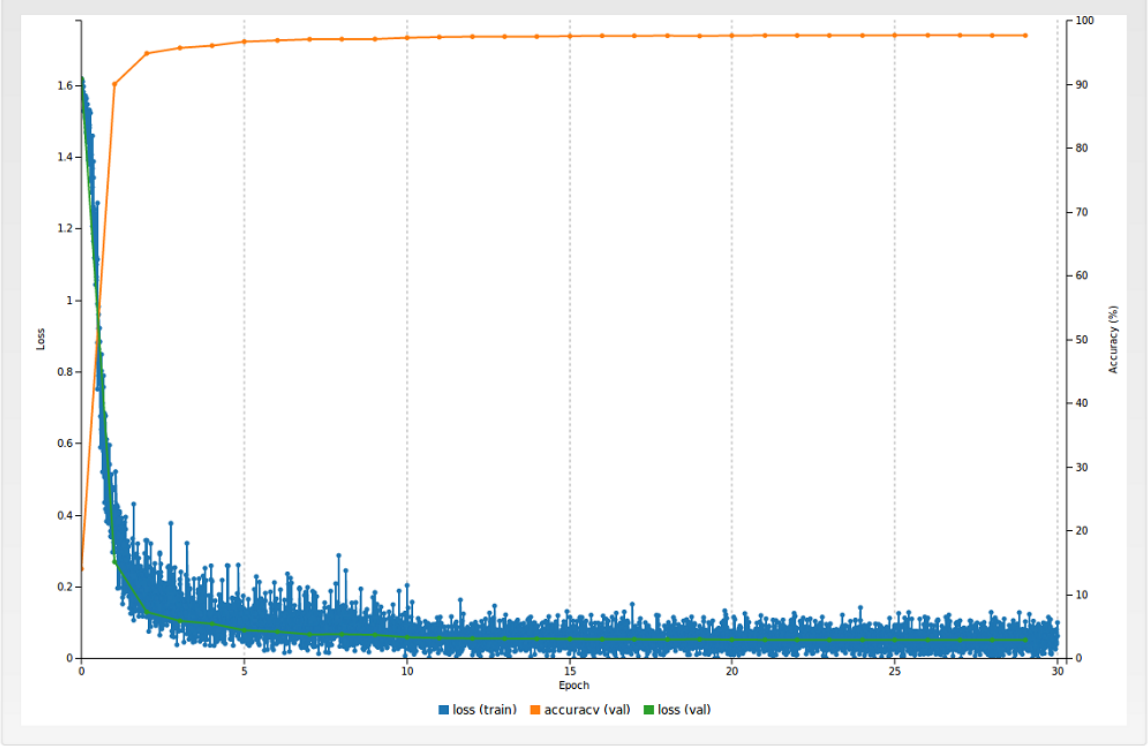


Figure 6.1: The accuracy and loss of the first experiment over 30 epochs.

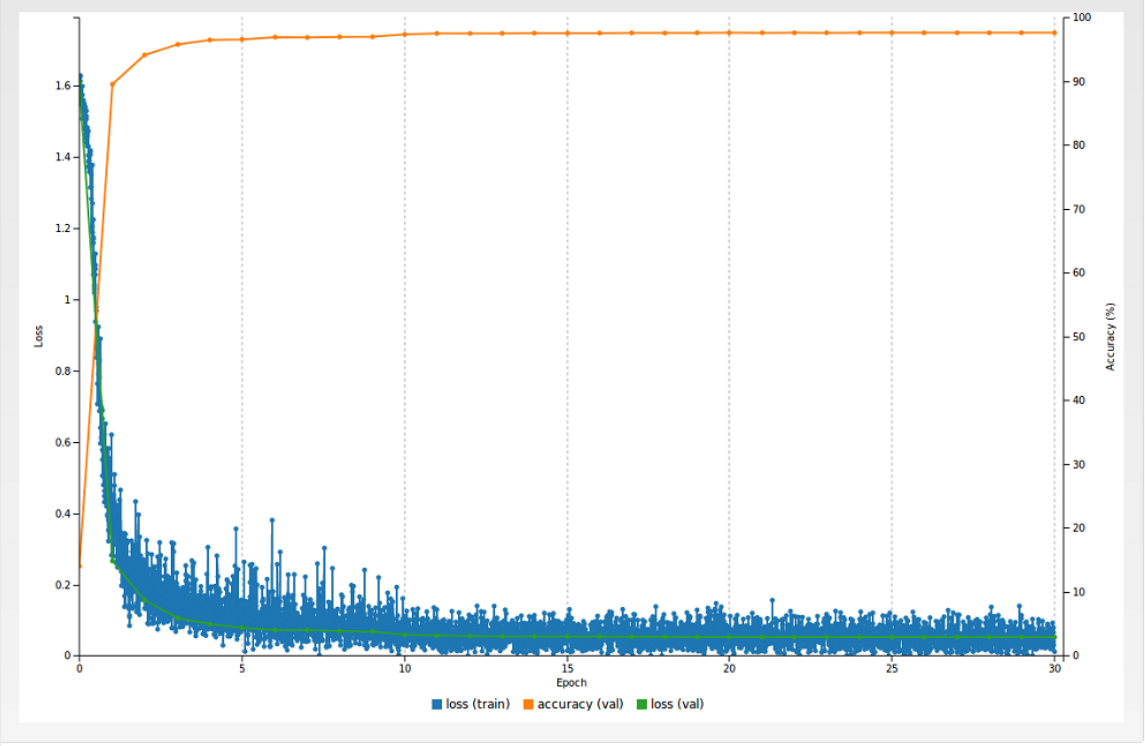


Figure 6.2: The accuracy and loss of the second experiment over 30 epochs.

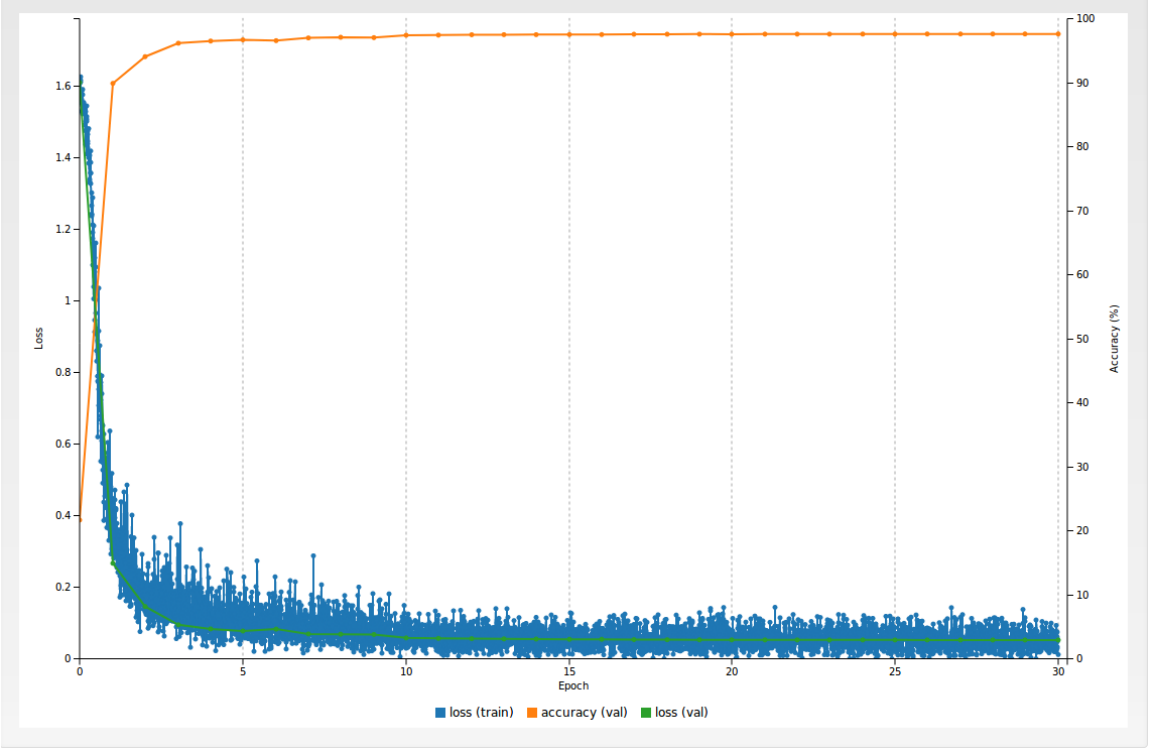


Figure 6.3: The accuracy and loss of the third experiment over 30 epochs.

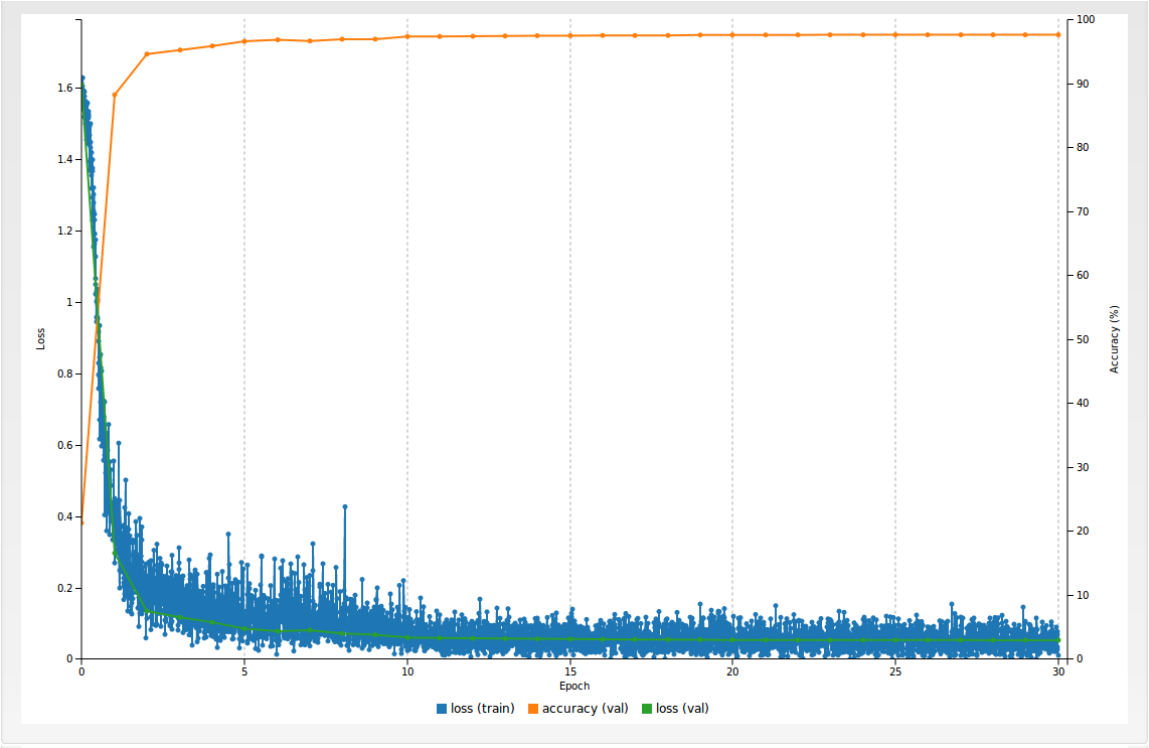


Figure 6.4: The accuracy and loss of the fourth experiment over 30 epochs.

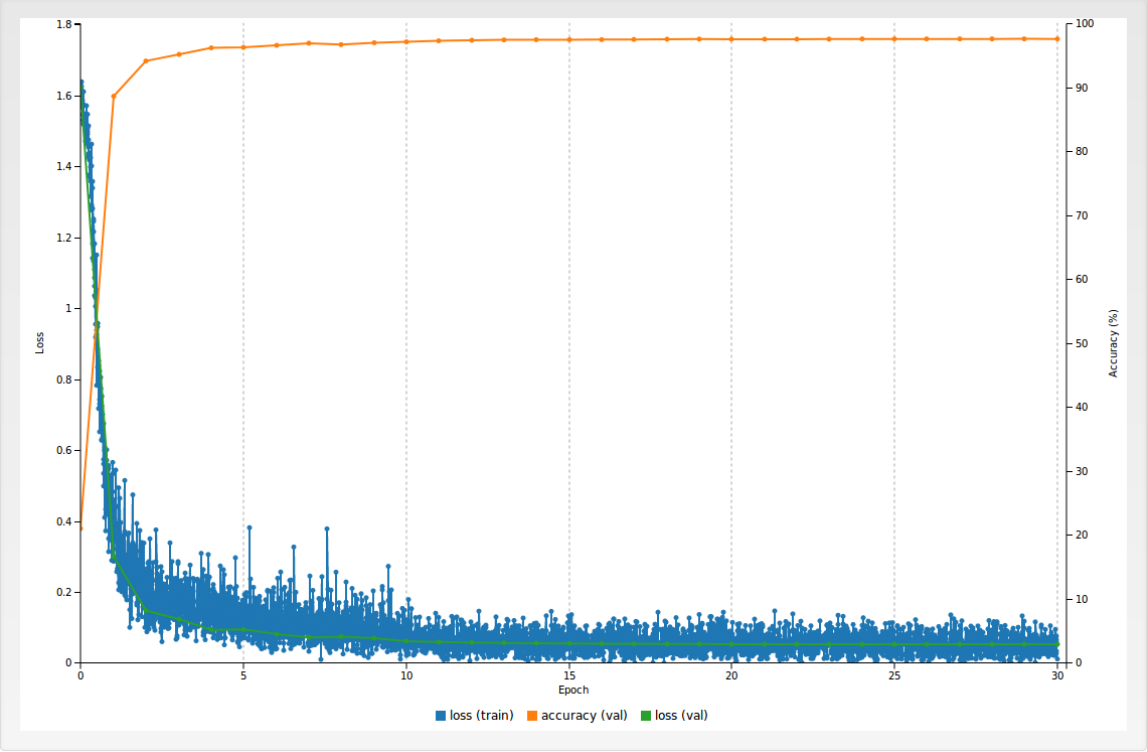


Figure 6.5: The accuracy and loss of the fifth experiment over 30 epochs.

6.1 Visualization of the Results

The results show that CNN architectures such as AlexNet are powerful models to classify different stages of Alzheimer's disease. In order to have a better understanding of the model and the convolutional layers, we visualized the weights filters and the statistical information of them in each layer of the model for many samples of our test dataset. The first visualization technique we used was monitoring the activations of the network during the forward pass. In the AlexNet model, which uses the ReLU method, the activations are more shapeless and dense at the beginning and they become more localized and sparse after the training progresses. This visualization can help us to check if some activation maps become all zero for different inputs. In the cases in which the activation maps became all zero we are facing a problem called "dead filters" which may be caused by high learning rates. This problem will result in a poor learning model and failure in classification.

In our model, since the first slices did not have enough functional information we noticed that activations have more zero values, which was expected. However, with enough functional information we monitored better activation behavior. To show sam-

ples of both successful and failed classification in the model, we chose two different images from our test dataset. The first sample (as failure in prediction) was the image of an SMC patient in the second slice and over a time course of 136, and is shown in Figure 6.6. After using the trained model on this sample the result of prediction was 28.18% NC, 22.31% LMCI, 21.61% EMCI, 17.63% AD, and 10.27%. Figure 6.7 shows the activation maps of this image in the first layer (left) and fifth layer (right).

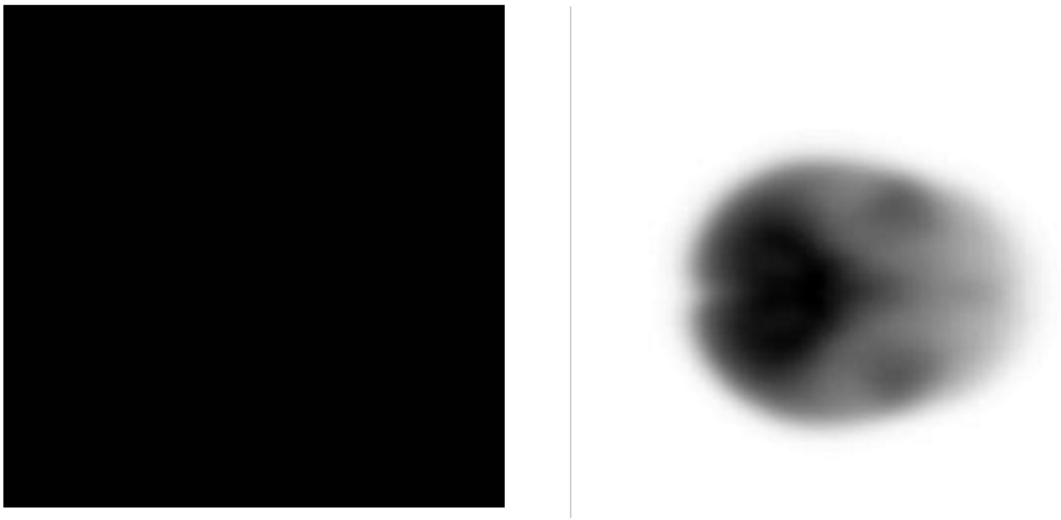


Figure 6.6: Second slice of fMRI scan on time course of 136 for a SMC patient (left). The activation data as an input for the model. The size of the filter applied in this part was 227×227.

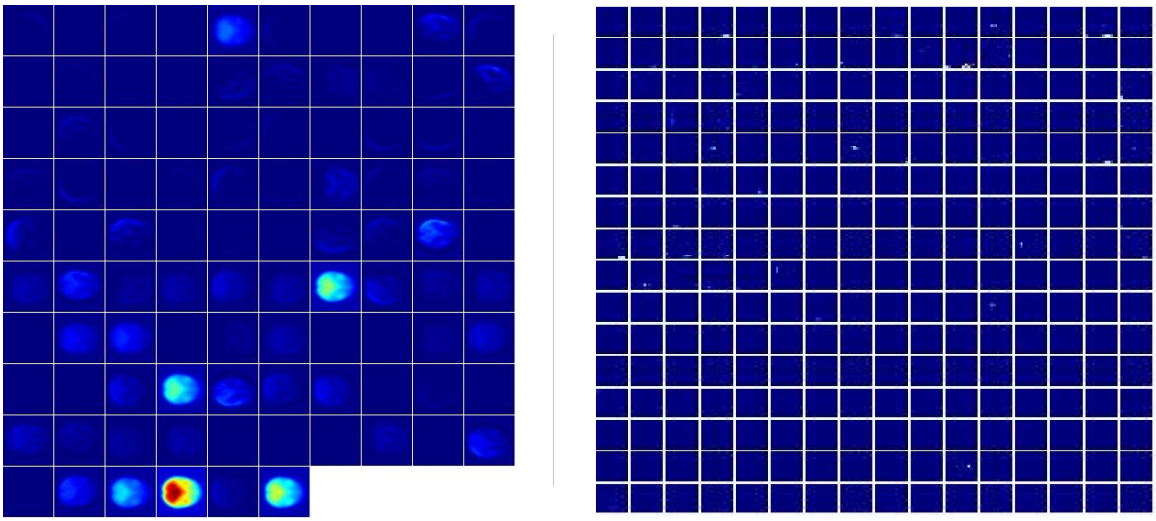


Figure 6.7: In the first CONV layer of AlexNet in a given trained model 96 filters of 55×55 pixels were applied and visualized for a SMC patients sample (left). In the fifth CONV layer of our model 256 filters of 13×13 were applied (right).

As seen in Figure 6.7, because of a lack of functional information in our data, the model failed to predict the result correctly. The visualization shows that the number of dead filters in this case is significant.

However, the rest of the results for most of the other images in our dataset show how our model is able to predict different stages with high accuracy.

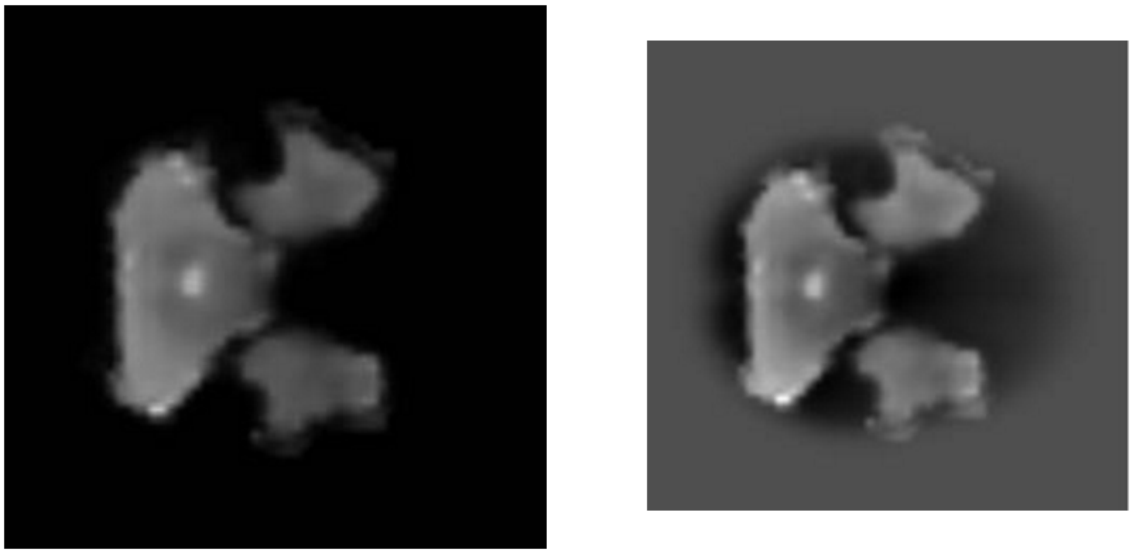


Figure 6.8: Slice number thirty two of an fMRI scan on a time course of 172 for a LMCI patient (left). The activation data as an input for the model. The size of the filter applied in this part was 227 × 227

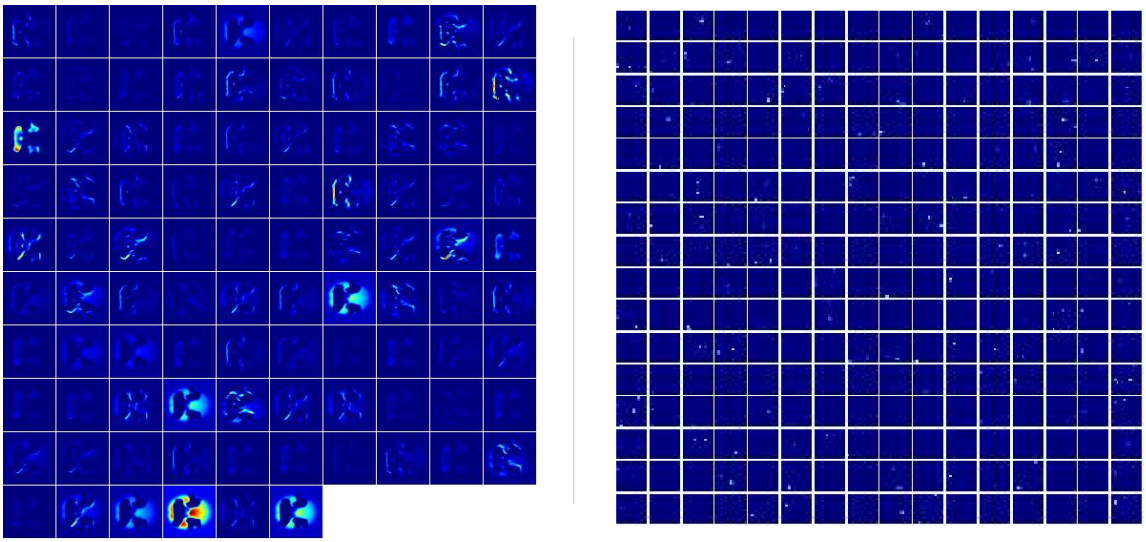


Figure 6.9: In the first CONV layer of AlexNet in a given trained model. 96 filters of 55×55 pixels were applied and visualized for a LMCI patient's sample (left). In the fifth CONV layer of our model 256 filters of 13 × 13 were applied (right).

Figure 6.8 is another example given to the trained model. In this case the image was in the LMCI category and the model predicted it as 100% LMCI. As can be seen in Figure 6.9, the number of dead filters in this case is reduced significantly and as a result we obtain a 100% accuracy in our prediction.

The second method to interpret the visualization of our CNN model is by visualizing the weights. As the first CONV layer of the network is working directly with raw data, we used this layer for our interpretation. Monitoring the weights of the network is useful since the less noisy and more smooth filters are, the more likely that our model trained in its best. Existence of noisy patterns is a sign that the model has not been trained enough or that there might be some over fitting in the model. Figure 6.10 shows a sample of successful prediction of LMCI patients.

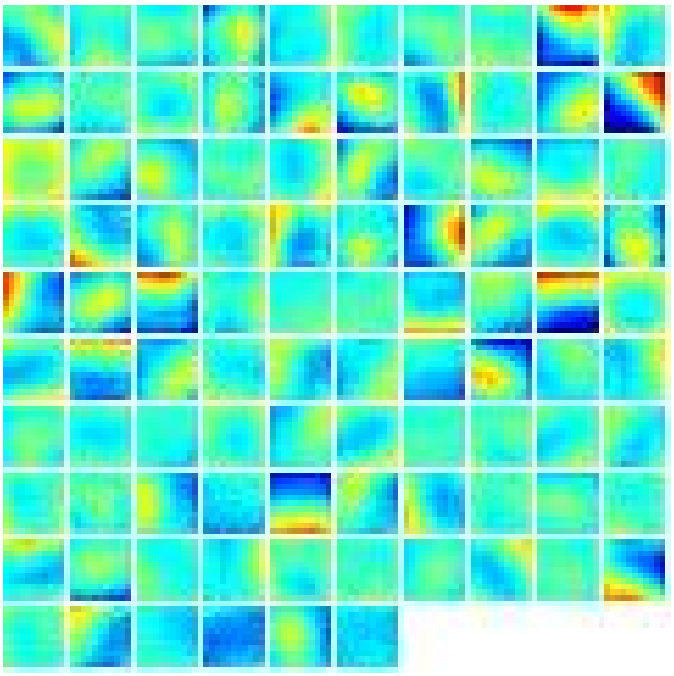


Figure 6.10: In the trained AlexNet model, 96 filters of size of 11 × 11 pixels were applied and visualized for the first layer of our network.

In our model all samples show smooth patterns without noise. This was an indication of acceptable performance of the trained model in our experiments.

The visualization of other layers in our models are shown in Appendix A and Appendix B. Appendix A provides an example of successful prediction for the LMCI class as shown in Figure 6.8, and Appendix B provides an example of unsuccessful prediction of the SMC class as shown in Figure 6.6.

6.2 ROC curves and accuracy of model for each class

The next step we considered in this thesis was studying the accuracy of our model for each class. For this purpose, we choose 264,000 images from our test datasets and applied the trained model. The result of this experiment was the accuracy of prediction of each class. For example, after giving an image from the EMCI category we obtained a 0.9837 probability that the image was classified as a member of the EMCI class, and the probabilities of the image being a member of other classes were $NC = 0.0082$, $LMCI = 0.0066$, $SMC = 0.0011$, and $AD = 0.0004$. We used this information to create a confusion matrix for all of the 264,000 testing images. The value of True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) were calculated for each class. Then, the accuracy of the performance of each class was calculated by the following formula:

$$Accuracy = \frac{TP}{TP+FN}. \tag{6.1}$$

	AD	EMCI	LMCI	NC	SMC	Accuracy per Class
AD	35148	159	386	1195	121	94.97%
EMCI	161	54130	327	1836	141	95.64%
LMCI	25	224	54809	2033	69	95.89%
NC	332	382	463	74276	76	98.34%
SMC	253	298	232	1271	35653	94.55%

Table 6.2: The confusion matrix for testing 264,000 images in our trained model.

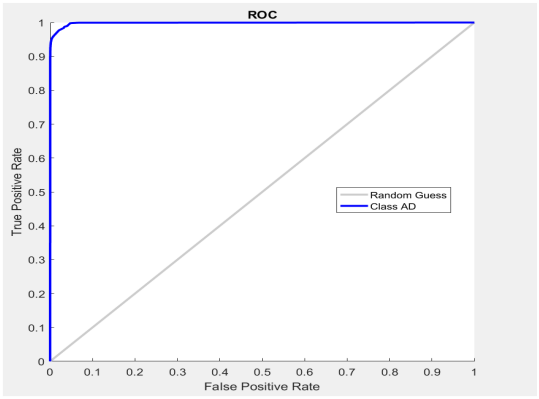
Table 6.2 shows the confusion matrix and accuracy calculated by Equation 6.1. The accuracy of the normal control class was higher. According to the number of images available in the normal control class, we expected to have a better performance in this class. This is because deep learning methods perform better with larger datasets.

Finally, the receiver operating characteristic (ROC) curves were calculated for each class to validate the performance of our model. For each class TP, FP, TN, and FN values were calculated for 100 thresholds from 0 to 1 using the MATLAB Neural Network toolbox [40]. To draw curves, the sensitivity and specificity values were calculated by the following equations:

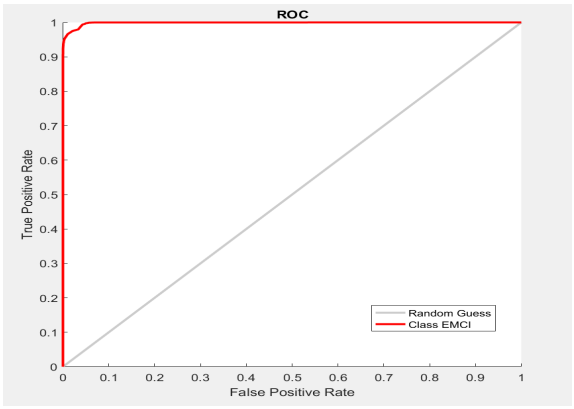
$$\begin{aligned} Sensitivity = Truepositiverate(TPR) &= \frac{TP}{(TP+FN)} \\ Specificity = Truenegativerate(TNR) &= \frac{TN}{(FP+TN)}. \end{aligned} \tag{6.2}$$

The axes of the ROC curve are sensitivity (True positive rate) and $1 - specificity$ (false positive rate).

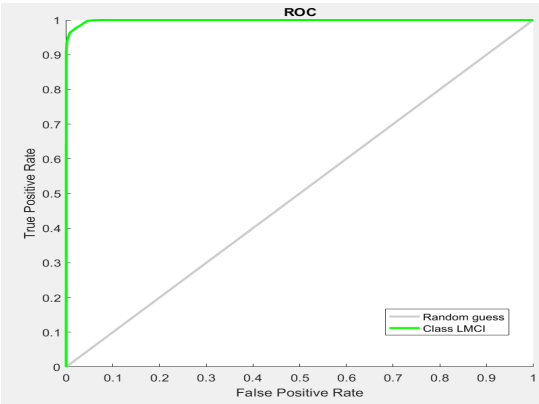
Figures 6.11a to 6.11e are illustrations of the ROC curves for each class. As shown, the performance of the model for each class is tending towards the ideal classification. Also, for all of the classes the performance was significantly better than a random guess.



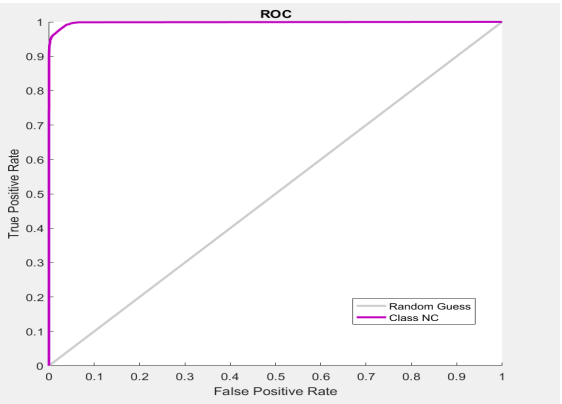
(a) Performance of the AD class



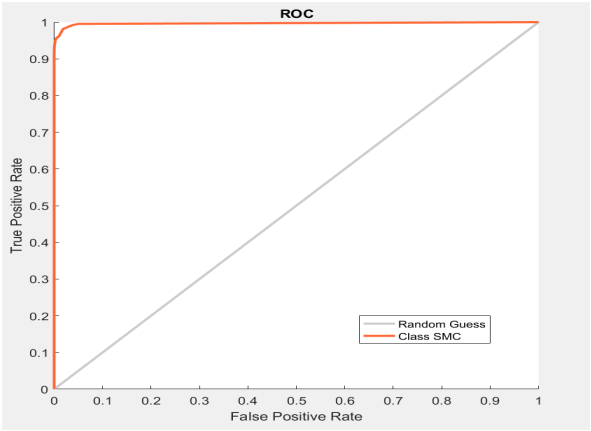
(b) Performance of the EMCI class



(c) Performance of the LMCI class



(d) Performance of the NC class



(e) Performance of the SMC class

Figure 6.11: ROC curves for evaluating the performance trained model for each class.

6.3 Comparison Between Our Experiment and Previous Works

As shown in Table 6.3, most previous work used binary classification between two classes of Alzheimer's disease and normal control subjects. All previous work which used either

Reference	Modality	Method	AD vs. MCI vs. NC	AD vs. NC	AD vs. MCI	NC vs. MCI	AD vs. NC vs. SMC vs. EMCI vs. LMCI
Suk et al. [60]	PET+MRI+CSF	SAE+SVM	N/A	95.9	N/A	85	N/A
Suk et al. [59]	PET+MRI	SAE+SVM	N/A	95.4	N/A	85.7	N/A
Zhu et al. [72]	PET+MRI+CSF	MSLF+SVM	N/A	95.9	N/A	82	N/A
Zu et al. [73]	PET+MRI	MTFS+SVM	N/A	96	N/A	80.3	N/A
Liu et al. [37]	PET+MRI	SAE+SVM	53.8	91.4	N/A	82.1	N/A
Liu et al. [36]	MRI	MFE+SVM	N/A	93.8	N/A	89.1	N/A
Li et al. [34]	PET+MRI+CSF	PCA+SVM	N/A	91.4	70.1	77.4	N/A
Payan et al. [46]	MRI	2D-SAE	89.4	95.4	86.8	92.1	N/A
Sarraf et al. [52] [51]	rs-fMRI	DL-CNN	N/A	99.9	N/A	N/A	N/A
Sarraf et al. [52] [51]	MRI	DL-CNN	N/A	100	N/A	N/A	N/A
Our Experiment	rs-fMRI	DL-CNN	N/A	N/A	N/A	N/A	97.64

Table 6.3: The comparison between our results and previous works based on the accuracy of the models.

multi-modal approaches or convolutional neural networks model obtained significant results in comparing AD vs. NC. However, their results in comparison between MCI vs. AD or MCI vs. NC were not as good. In our experiment we applied the existing deep learning models which shows a significant improvement compared to other methods to this problem, to classify five different classes of Alzheimer's disease. Besides, as shown in the previous section (Table 6.2), the accuracy of each class improved compared to previous methods. We also calculated the area under the curve (AUC) of the ROC curves. For SMC class the AUC value was 0.9334, for NC class it was 0.9486, 0.9500 for LMCI class, 0.9491 for EMCL class, and finally the AUC value calculated for AD class was 0.9422. This shows the consistency of our results.

Chapter 7

Conclusion and Future Work

In this work, we used strict preprocessing steps on raw fMRI data obtained from the ADNI dataset, and applied the AlexNet CNN classifier on preprocessed data to classify different stages of Alzheimer's disease (NC, SMC, EMCI, LMCI, and AD). The low to high level features were learned during classification, resulting in an average accuracy of 97.64%. This is an outstanding result, compared to applications of other methods for classifying different stages of Alzheimer's disease. Besides, previous classification methods were applied to either AD vs. NC as binary classification or with three classes of NC vs. MCI vs. AD. This is, to the best of our knowledge, the first time that a deep learning algorithm has been applied to classifying Alzheimer's disease according to all other available stages.

The goal of this thesis was classification of all stages of Alzheimer's disease which was not done in previous work. The availability of this data from the ADNI project will give us the opportunity to use the previous multi modal methods [34] [37] [36] [72] [73] [59] [60] with the assistance of SVM techniques to improve the accuracy of classification in this problem.

As the deep learning models are improving everyday and are applied in different fields, it is possible to use new and more advanced models such as VGG, GoogleNet, ResNet, etc and compare the results. Besides, applying pre-trained models and transfer learning may show even better results, as pre-trained models have shown outstanding results in medical imaging studies [63]. In deep learning, especially if the model has more layers and parameters, more data results in better performance. Therefore, increasing the size of the dataset is an important factor in improving the performance of the training model.

Last but not least, the studies show that deep learning approaches were successful in different areas of neuroimaging. The application of new models on fMRI, MRI, and

PET dataset for different brain disorders and diseases such as Autism, Attention Deficit Hyperactivity Disorder (ADHD), Schizophrenia, and Huntington's disease are new case studies in the state of the art of deep learning models [13] [48]. In addition, using task based fMRI data has been used to understand the brain development process such as studying language development in children [6]. Also, using task based fMRI to study diseases and the effects of brain disorders on the performance of the brain is an important approach to help physicians in terms of studying disease progress. The application of deep learning in task based fMRI data is a valuable case study to understand the brain development process.

Bibliography

[1] ADNI. Background and Rationale, http://adni.loni.usc.edu/study-design/background-rationale/, 2017.

[2] Alzheimer's association. 2017 Alzheimer's disease facts and figures. Special Report, $https://www.alz.org/documents_custom/2017-facts-and-figures.pdf$.

[3] I Arel, D C Rose, and T P Karnowski. Deep Machine Learning - A New Frontier in Artificial Intelligence Research [Research Frontier]. *IEEE Computational Intelligence Magazine*, 5(4):13–18, 11 2010.

[4] Peter Bandettini. fMRI Summer Course 2016, $https://fmrif.nimh.nih.gov/course/2016/01_Bandettini_20160531$, 2016.

[5] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006.

[6] D V M Bishop. How does the brain learn language? Insights from the study of children with and without language impairment. *Developmental medicine and child neurology.*, 42(2):133–142, 2000.

[7] Nicolas Brunel, Vincent Hakim, and Magnus JE Richardson. Single neuron dynamics and computation. *Current Opinion in Neurobiology*, 25:149–155, 4 2014.

[8] Joshua Carp, Joonkoo Park, Thad A. Polk, and Denise C. Park. Age differences in neural distinctiveness revealed by multi-voxel pattern analysis. *NeuroImage*, 56(2):736–743, 5 2011.

[9] Guillaume Chanel, Swann Pichon, Laurence Conty, Sylvie Berthoz, Coralie Chevallier, and Julie Grèzes. Classification of autistic individuals and controls using cross-task characterization of fMRI activity. *NeuroImage. Clinical*, 10:78–88, 2016.

[10] Roden Chris. NITRC: dcm2nii: File Release Notes and Changelog, $https://www.nitrc.org/frs/shownotes.php?release_id=3336$.

[11] Wikimedia Commons. File:ANN neuron.svg - Wikimedia Commons, $https://commons.wikimedia.org/w/index.php?title=File:ANN_neuron.svg\&oldid=125527801$.

[12] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine Learning*, 20(3):273–297, 9 1995.

[13] Marc N. Coutanche, Sharon L. Thompson-Schill, and Robert T. Schultz. Multi-voxel pattern analysis of fMRI data predicts clinical symptom severity. *NeuroImage*, 57(1):113–123, 2011.

[14] Federico De Martino, Giancarlo Valente, NoÃńl Staeren, John Ashburner, Rainer Goebel, and Elia Formisano. Combining multivariate voxel selection and support vector machines for mapping and classification of fMRI spatial patterns. *NeuroImage*, 43(1):44–58, 2008.

[15] Thomas Eckert, Anna Barnes, Vijay Dhawan, Steve Frucht, Mark F. Gordon, Andrew S. Feigin, and D. Eidelberg. FDG PET in the differential diagnosis of parkinsonian disorders. *NeuroImage*, 26(3):912–921, 7 2005.

[16] Laurene V. Fausett. *Fundamentals of neural networks : architectures, algorithms, and applications*. Prentice-Hall, 1994.

[17] Serena Yeung Fei-Fei Li, Justin Johnson. CS231n Convolutional Neural Networks for Visual Recognition, 2017.

[18] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep Sparse Rectifier Neural Networks. In *the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 315–323, 6 2011.

[19] Irene Tracey Hannah Devlin, Heidi Johansen-Berg, and Stuart Clare. Introduction to FMRI, https://www.ndcn.ox.ac.uk/divisions/fmrib/what-is-fmri/introduction-to-fmri.

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 12 2016. IEEE.

[21] Geoffrey E. Hinton, Simon Osindero, and Yee-Whye Teh. A Fast Learning Algorithm for Deep Belief Nets. *Neural Computation*, 18(7):1527–1554, 7 2006.

[22] Geoffrey E. Hinton and Russle R. Salakhutdinov. Reducing the Dimensionality of Data with Neural Networks. *Science*, 313(5786):504–507, 7 2006.

[23] Geoffrey E. Hinton, Nitish Srivastava, Alex Krizhevsky, Ilya Sutskever, and Ruslan R. Salakhutdinov. Improving neural networks by preventing co-adaptation of feature detectors. 7 2012.

[24] William J. Jagust, Dan Bandy, Kewei Chen, Norman L. Foster, Susan M. Landau, Chester A. Mathis, Julie C. Price, Eric M. Reiman, Daniel Skovronsky, Robert A. Koeppe, and Alzheimer's Disease Neuroimaging Initiative. The Alzheimer's Disease Neuroimaging Initiative positron emission tomography core. *Alzheimer's & Dementia*, 6(3):221–229, 5 2010.

[25] M Jenkinson and S Smith. A global optimisation method for robust affine registration of brain images. *Medical image analysis*, 5(2):143–56, 6 2001.

[26] Mark Jenkinson, Peter Bannister, Michael Brady, and Stephen Smith. Improved Optimization for the Robust and Accurate Linear Registration and Motion Correction of Brain Images. *NeuroImage*, 17(2):825–841, 10 2002.

[27] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. Caffe: Convolutional Architecture for Fast Feature Embedding. In *Multimedia*, pages 675–678. ACM, 6 2014.

[28] Jimmy Shen. Tools for NIfTI and ANALYZE image - File Exchange - MATLAB Central, https://www.mathworks.com/matlabcentral/fileexchange/8797-tools-for-nifti-and-analyze-image, 2014.

[29] Keith A Johnson, Nick C Fox, Reisa A Sperling, and William E Klunk. Brain imaging in Alzheimer disease. *Cold Spring Harbor perspectives in medicine*, 2(4):a006213, 4 2012.

[30] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[31] S. LaConte, S. Strother, V. Cherkassky, J. Anderson, and X. Hu. Support vector machines for temporal classification of block design fMRI data. *NeuroImage*, 26(2):317–329, 6 2005.

[32] Y. LeCun, L. Jackel, L. Bottou, A. Brunot, C. Cortes, J. Denker, H. Drucker, I. Guyon, U. Müller, E. Säckinger, P. Simard, and V. Vapnik. Comparison of Learning Algorithms for Handwritten Digit Recognition. *ICANN*, pages 53–60, 1995.

[33] T.M. Lehmann, C. Gonner, and K. Spitzer. Survey: interpolation methods in medical image processing. *IEEE Transactions on Medical Imaging*, 18(11):1049–1075, 1999.

[34] Feng Li, Loc Tran, Kim-Han Thung, Shuiwang Ji, Dinggang Shen, and Jiang Li. A Robust Deep Model for Improved Classification of AD/MCI Patients. *EEE Journal of Biomedical and Health Informatics*, 19(5):1610–1616, 9 2015.

[35] Martin A Lindquist. The Statistical Analysis of fMRI Data. *Statistical Science*, 23(4):439–464, 2008.

[36] Mingxia Liu, Daoqiang Zhang, Ehsan Adeli, and Dinggang Shen. Inherent Structure-Based Multiview Learning With Multitemplate Feature Representation for Alzheimer's Disease Diagnosis. *IEEE transactions on bio-medical engineering*, 63(7):1473–82, 7 2016.

[37] Siqi Liu, Sidong Liu, Weidong Cai, Hangyu Che, Sonia Pujol, Ron Kikinis, Dagan Feng, Michael J. Fulham, and ADNI. Multimodal Neuroimaging Feature Learning for Multiclass Diagnosis of Alzheimer's Disease. *IEEE Transactions on Biomedical Engineering*, 62(4):1132–1140, 4 2015.

[38] Siqi Liu, Sidong Liu, Weidong Cai, Sonia Pujol, Ron Kikinis, and Dagan Feng. Early diagnosis of Alzheimer's disease with deep learning. In *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, pages 1015–1018. IEEE, 4 2014.

[39] Michael London and Michael Häusser. Dendritic Computation. *Annual review of neuroscience*, 58, 2005.

[40] Mathworks. Neural Network Toolbox - MATLAB: (R2016b), https://www.mathworks.com/products/neural-network.html, 2016.

[41] Martin M Monti. Statistical Analysis of fMRI Time-Series: A Critical Review of the GLM Approach. *Frontiers in human neuroscience*, 5:28, 2011.

[42] Henson R. N.A. Analysis of fMRI time series: Linear Time-Invariant models, event-related fMRI and optimal experimental design. In *event-related fMRI and optimal experimental design*, pages 793–822. Elsevier: London, 2003.

[43] National Institute on aging. Alzheimer's Disease Fact Sheet, https://www.nia.nih.gov/health/alzheimers-disease-fact-sheet.

[44] S Ogawa, D W Tank, R Menon, J M Ellermann, S G Kim, H Merkle, and K Ugurbil. Intrinsic signal changes accompanying sensory stimulation: functional brain mapping with magnetic resonance imaging. *Proceedings of the National Academy of Sciences of the United States of America*, 89(13):5951–5, 7 1992.

[45] Mohammad Ali Oghabian, Seyed Amir Hossein Batouli, Maryam Norouzian, Maryam Ziaei, and Hajir Sikaroodi. Using functional Magnetic Resonance Imaging to differentiate between healthy aging subjects, Mild Cognitive Impairment, and Alzheimer's patients. *Journal of research in medical sciences : the official journal of Isfahan University of Medical Sciences*, 15(2):84–93, 3 2010.

[46] Adrien Payan and Giovanni Montana. Predicting Alzheimer's disease: a neuroimaging study with 3D convolutional neural networks. In *ICPRAM*, 2 2015.

[47] Francisco Pereira, Tom Mitchell, and Matthew Botvinick. Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, 45(1):S199–S209, 3 2009.

[48] Sergey M Plis, Devon R Hjelm, Ruslan Salakhutdinov, Elena A Allen, Henry J Bockholt, Jeffrey D Long, Hans J Johnson, Jane S Paulsen, Jessica A Turner, and Vince D Calhoun. Deep learning for neuroimaging: a validation study. *Frontiers in neuroscience*, 8:229, 1 2014.

[49] Shannon L Risacher and Andrew J Saykin. Neuroimaging and other biomarkers for Alzheimer's disease: the changing landscape of early detection. *Annual review of clinical psychology*, 9:621–48, 2013.

[50] Saman Sarraf and Jian Sun. Functional Brain Imaging: A Comprehensive Survey. 2 2016.

[51] Saman Sarraf and Ghassem Tofighi. Classification of Alzheimer's Disease using fMRI Data and Deep Learning Convolutional Neural Networks. *CoRR*, abs/1603.0, 3 2016.

[52] Saman Sarraf and Ghassem Tofighi. Deep Learning-based Pipeline to Recognize AlzheimerâĂšs Disease using fMRI Data. In *uture Technologies Conference (FTC)*. IEEE, 2016.

[53] Toby Segaran. *Programming collective intelligence : building smart web 2.0 applications*. O'Reilly Media, 2007.

[54] Hoo-Chang Shin, Holger R Roth, Mingchen Gao, Le Lu, Ziyue Xu, Isabella Nogues, Jianhua Yao, Daniel Mollura, and Ronald M Summers. Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning. *IEEE transactions on medical imaging*, 35(5):1285–98, 2016.

[55] Karen Simonyan and Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *ICLR*, 9 2015.

[56] Vikram Singh. Convolutional Neural Networks for Image Classification, https://www.completegate.com/2017022864/blog/deep-machine-learning-images-lenet-alexnet-cnn/all-pages, 2017.

[57] Stephen M. Smith. Fast robust automated brain extraction. *Human Brain Mapping*, 17(3):143–155, 11 2002.

[58] Rainulf A Stelzma, H Norman Schnitzlein, and F Reed Murllagh. An English Translation of Alzheimer's 1907 Paper, "Uber eine eigenartige Erlranliung der Hirnrinde". *Clinical Anatomy*, 8:429–43, 1995.

[59] Heung-Il Suk, Seong-Whan Lee, Dinggang Shen, and the Alzheimers Disease Neuroimaging Alzheimer's Disease Neuroimaging Initiative. Hierarchical feature representation and multimodal fusion with deep learning for AD/MCI diagnosis. *NeuroImage*, 101:569–82, 11 2014.

[60] Heung-Il Suk and Dinggang Shen. Deep learning-based feature representation for AD/MCI classification. *Medical image computing and computer-assisted intervention : MICCAI ... International Conference on Medical Image Computing and Computer-Assisted Intervention*, 16(Pt 2):583–90, 2013.

[61] H. N. Suma and S. Murali. Pattern Classification and Analysis of Brain Maps through fMRI data with Multiple Methods. *International Journal of Computer Applications*, 1(27):117–126, 2 2010.

[62] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going Deeper with Convolutions. In *Computer Vision and Pattern Recognition (CVPR)*. IEEE, 9 2015.

[63] Nima Tajbakhsh, Jae Y. Shin, Suryakanth R. Gurudu, R. Todd Hurst, Christopher B. Kendall, Michael B. Gotway, and Jianming Liang. Convolutional Neural Networks for Medical Image Analysis: Full Training or Fine Tuning? *IEEE Transactions on Medical Imaging*, 35(5):1299 – 1312, 5 2016.

[64] Rik Vandenberghe, Koen Van Laere, Adrian Ivanoiu, Eric Salmon, Christine Bastin, Eric Triau, Steen Hasselbalch, Ian Law, Allan Andersen, Alex Korner, Lennart Minthon, GaÃńtan Garraux, Natalie Nelissen, Guy Bormans, Chris Buckley, Rikard Owenius, Lennart Thurfjell, Gill Farrar, and David J. Brooks. 18F-flutemetamol amyloid imaging in Alzheimer disease and mild cognitive impairment: A phase 2 trial. *Annals of Neurology*, 68(3):319–329, 9 2010.

[65] Prashanthi Vemuri, David T Jones, and Clifford R Jack. Resting state functional MRI in Alzheimer's Disease. *Alzheimer's research & therapy*, 4(1):2, 1 2012.

[66] Heather C. Whalley, Enrico Simonotto, William Moorhead, Andrew McIntosh, Ian Marshall, Klaus P. Ebmeier, David G.C. Owens, Nigel H. Goddard, Eve C. Johnstone, and Stephen M. Lawrie. Functional Imaging as a Predictor of Schizophrenia. *Biological Psychiatry*, 60(5):454–462, 9 2006.

[67] Wikimedia Commons. File:Typical cnn.png - Wikimedia Commons, $https://commons.wikimedia.org/w/index.php?title=File:Typical_cnn.png\&oldid=204200221$.

[68] Wikimedia Commons Contributors. File:Alzheimer's disease brain comparison.jpg,$https://commons.wikimedia.org/wiki/File:Alzheimer\%27s_disease_brain_comparison.jpg$.

[69] D. F. Wong, P. B. Rosenberg, Y. Zhou, A. Kumar, V. Raymont, H. T. Ravert, R. F. Dannals, A. Nandi, J. R. Brasic, W. Ye, J. Hilton, C. Lyketsos, H. F. Kung, A. D. Joshi, D. M. Skovronsky, and M. J. Pontecorvo. In Vivo Imaging of Amyloid Deposition in Alzheimer Disease Using the Radioligand 18F-AV-45 (Flobetapir F 18). *Journal of Nuclear Medicine*, 51(6):913–920, 6 2010.

[70] Tony Wyss-Coray. Ageing, neurodegeneration and brain rejuvenation. *Nature*, 539(7628):180–186, 2016.

[71] Matthew D Zeiler and Rob Fergus. Visualizing and Understanding Convolutional Networks. In *Computer Vision âĂŞ ECCV*, pages 818–833. Springer, 11 2014.

[72] Xiaofeng Zhu, Heung-Il Suk, and Dinggang Shen. A novel matrix-similarity based loss function for joint regression and classification in AD diagnosis. *NeuroImage*, 100:91–105, 10 2014.

[73] Chen Zu, Biao Jie, Mingxia Liu, Songcan Chen, Dinggang Shen, and Daoqiang Zhang. Label-aligned multi-task feature learning for multimodal classification of Alzheimer's disease and mild cognitive impairment. *Brain Imaging and Behavior*, 10(4):1148–1159, 12 2016.

Appendix A

Visualization results of different layers of AlexNet model for a sample of successful classification

In this Appendix we show the visualization of different layers of model for a patient in LMCI class which was classified into the correct category.

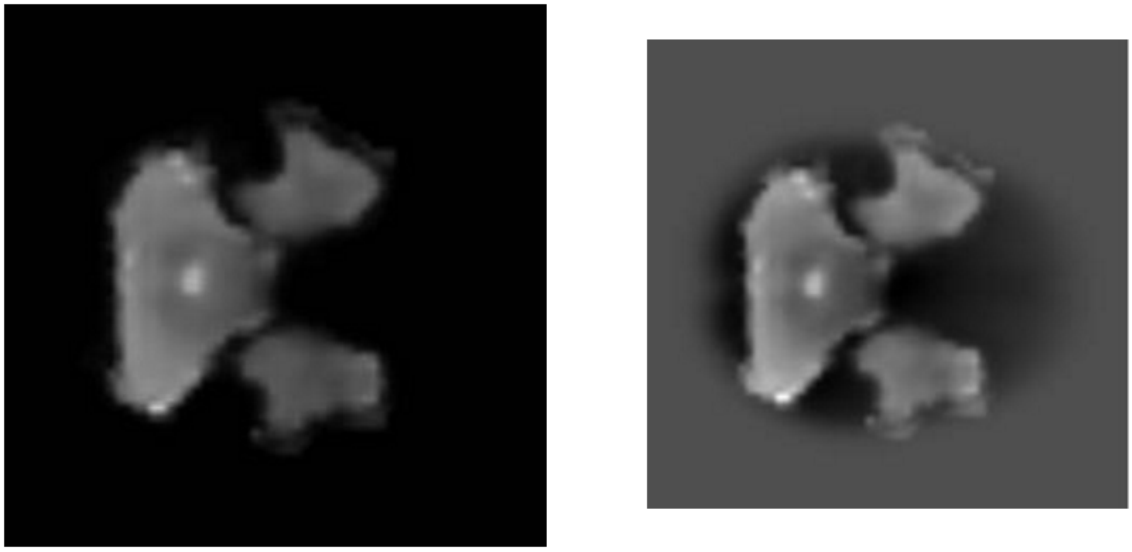


Figure A.1: Second slice of fMRI scan on time course of 136 for a LMCI patient (left). The activation data as an input for the model. The size of filter applied in this part was 227×227.

Class	prediction
LMCI	100%
AD	0.00%
EMCI	0.00%
NC	0.00%
SMC	0.00%

Table A.1: The prediction of the model for a sample image belong to LMCI category.

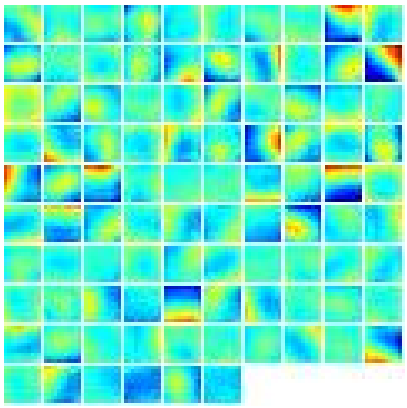


Figure A.2: In the trained AlexNet model, 96 filters of size of 11 × 11 pixels were applied and visualized for the first convolution layer of our network.

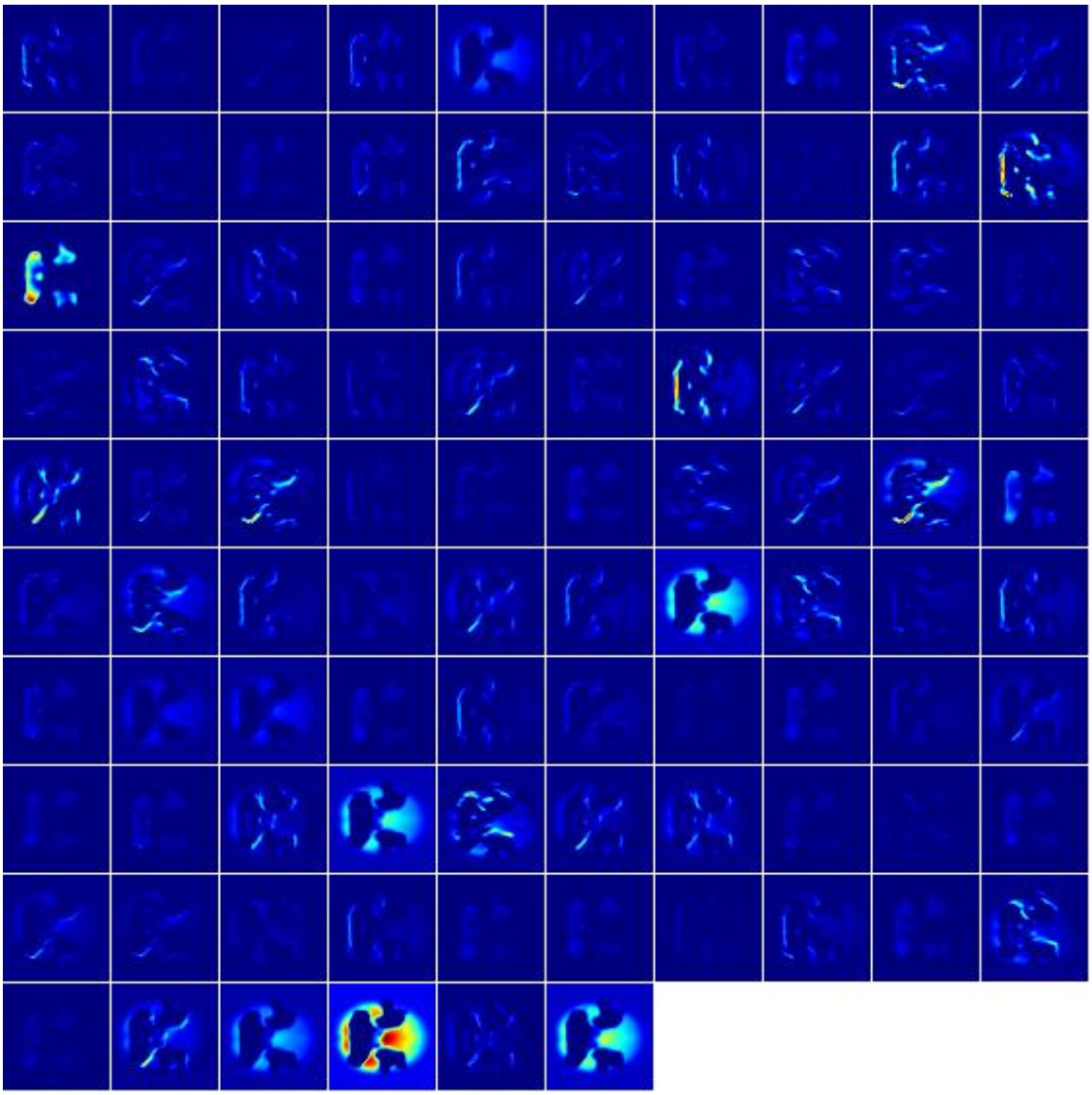


Figure A.3: In the first CONV layer of AlexNet in a given trained model 96 filters of 55×55 pixels were applied and visualized for a SMC patients sample. This images is a visualization of the first CONV layer after applying the activation function.

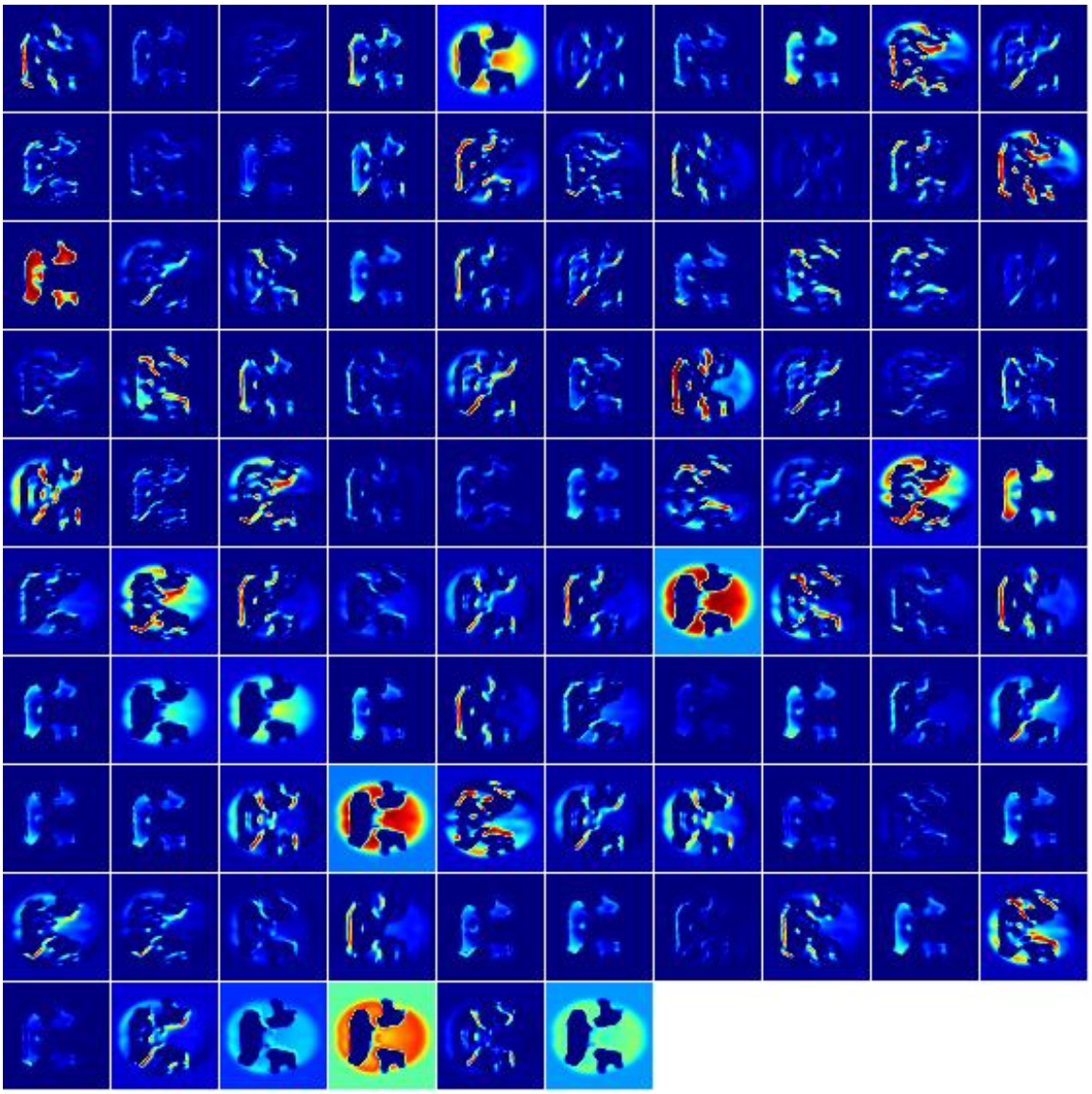


Figure A.4: The visualization of the first CONV layer after applying the activation function and normalization.

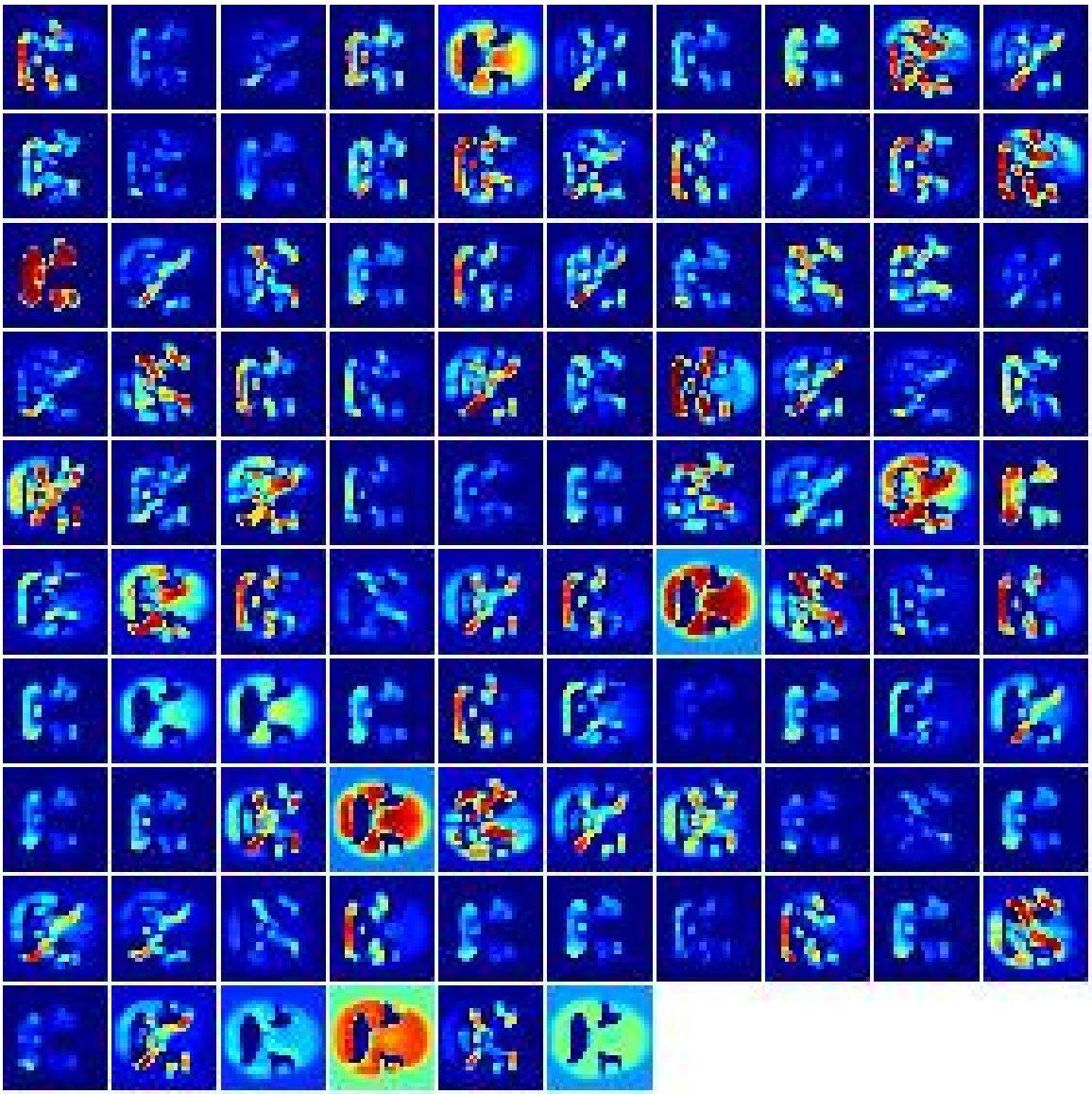


Figure A.5: The visualization of the first pooling layer with 96 filters of size 27 × 27 pixels
.

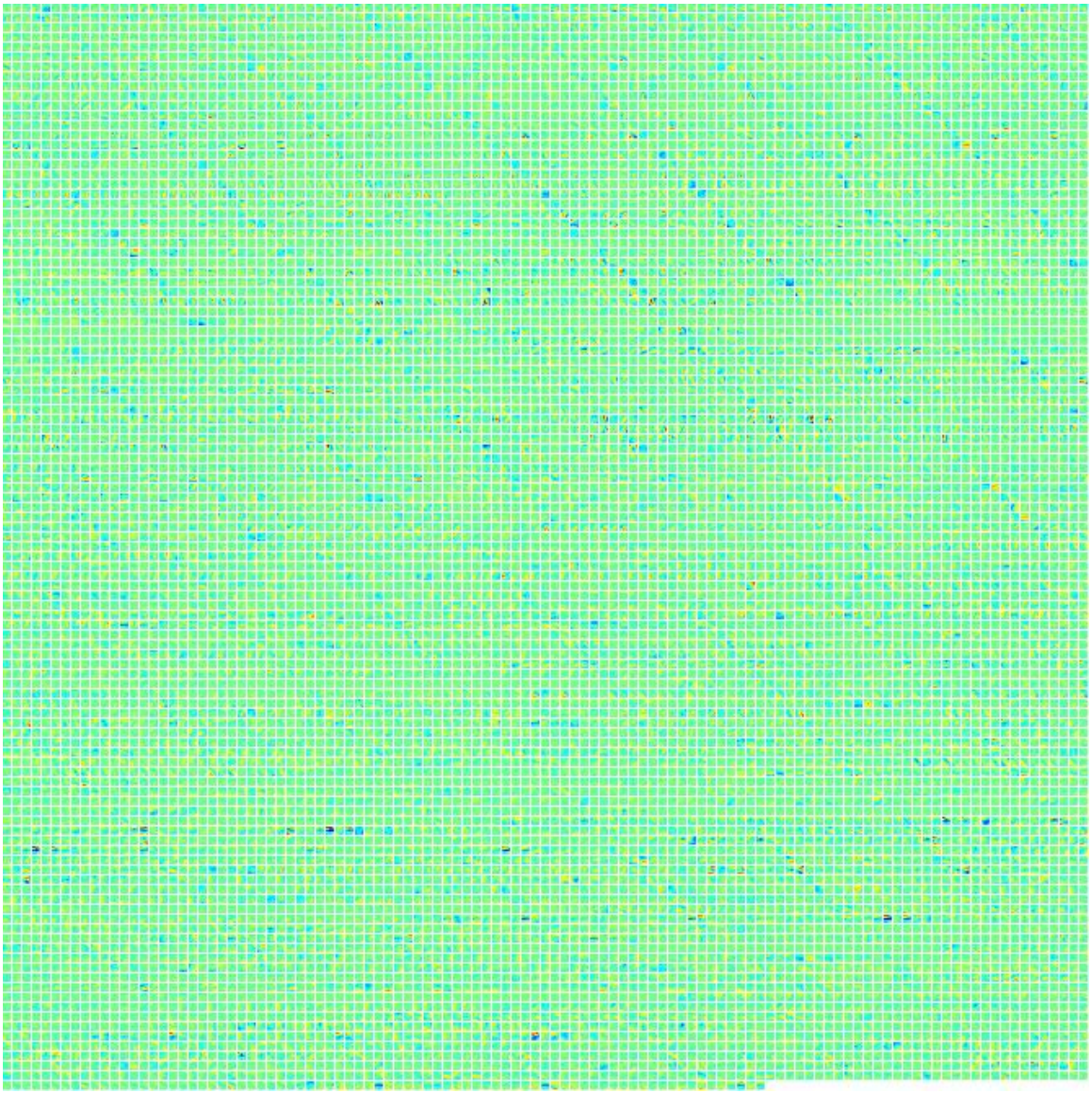


Figure A.6: In the trained AlexNet model, 256 filters of size of 5 × 5 pixels were applied and visualized for the second convolution layer of our network.

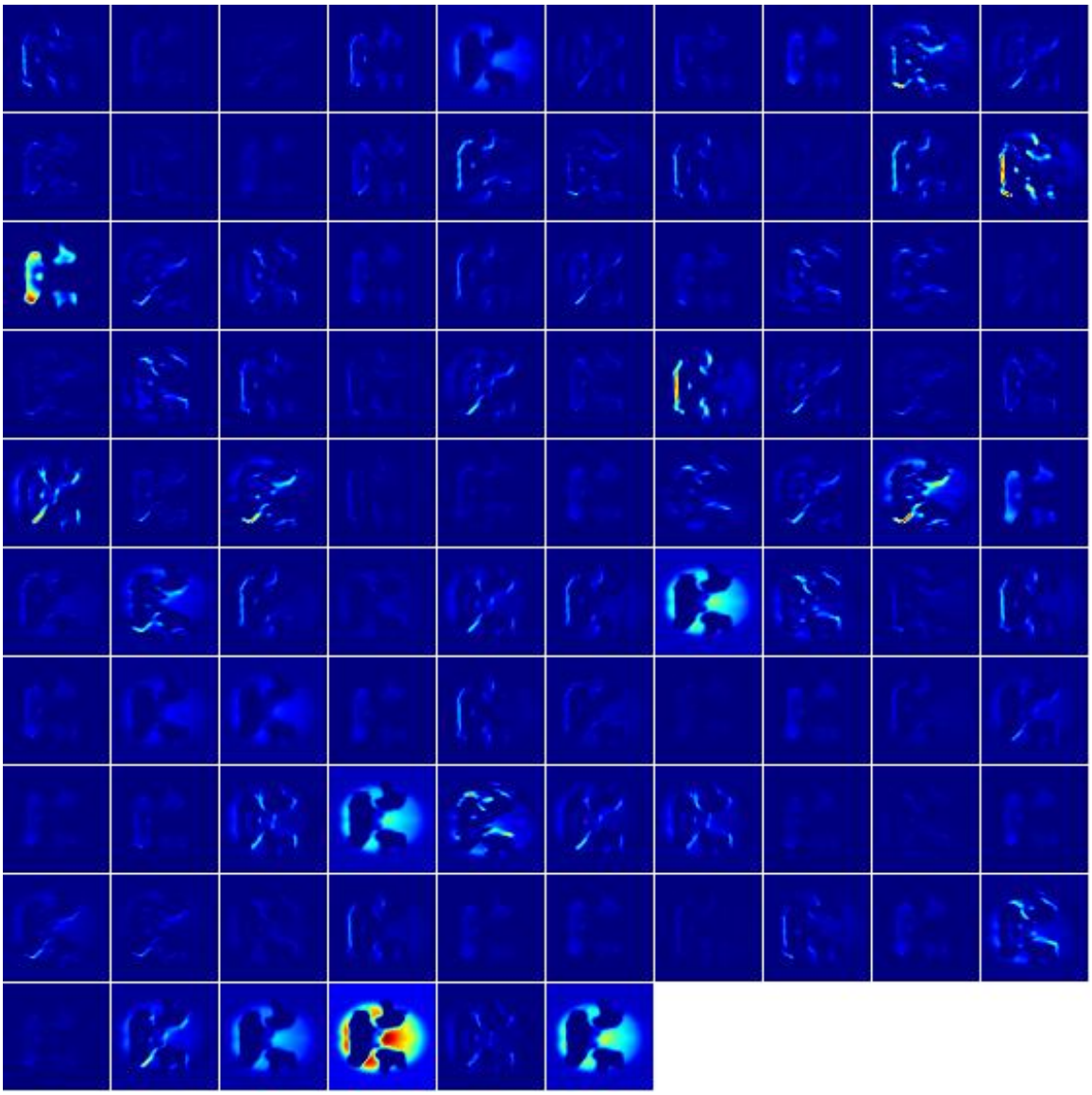


Figure A.7: In the second CONV layer of AlexNet in a given trained model 256 filters of 27×27 pixels were applied and visualized for a LMCI patients sample.

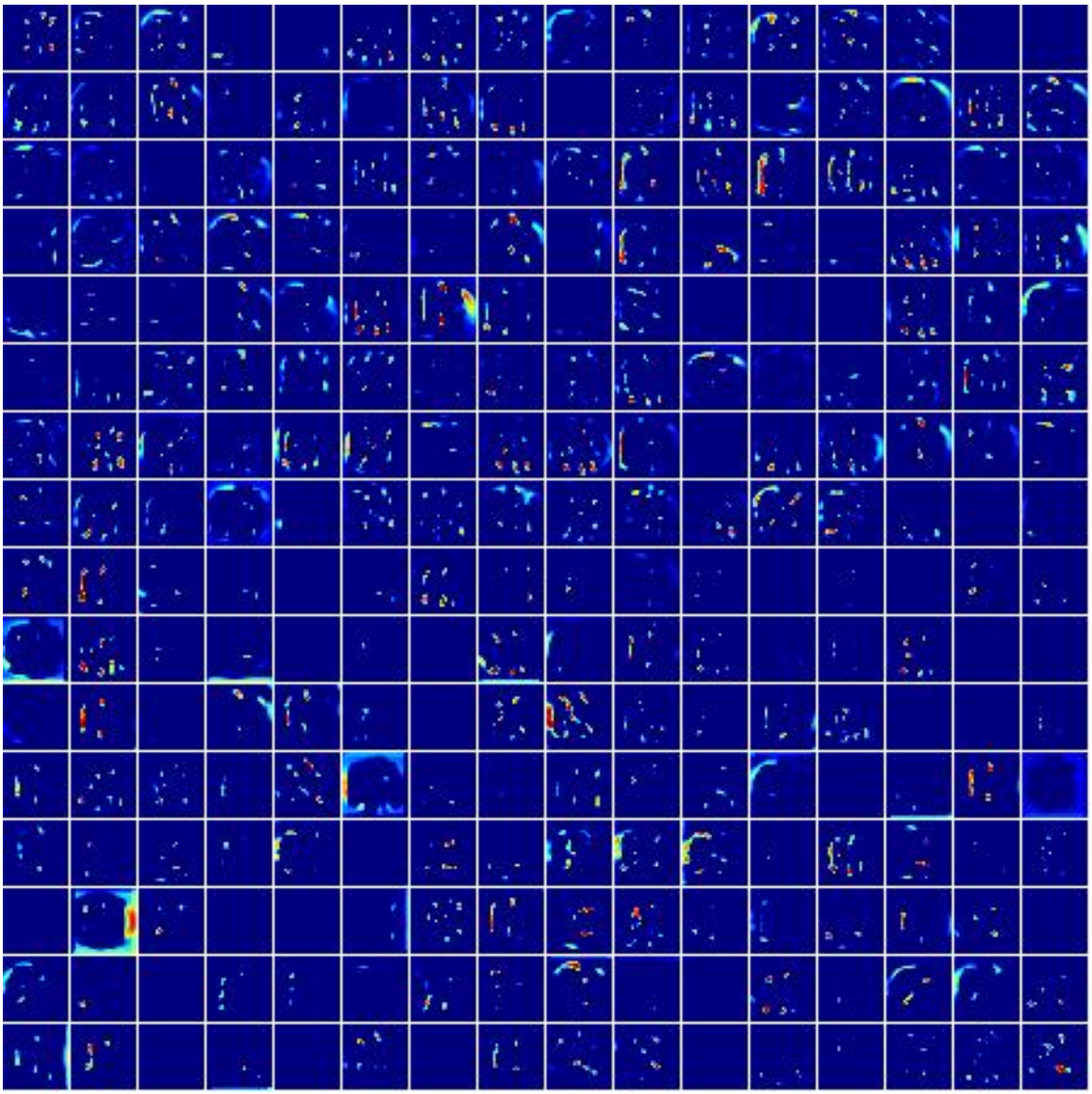


Figure A.8: The visualization of the second CONV layer after applying the activation function and normalization.

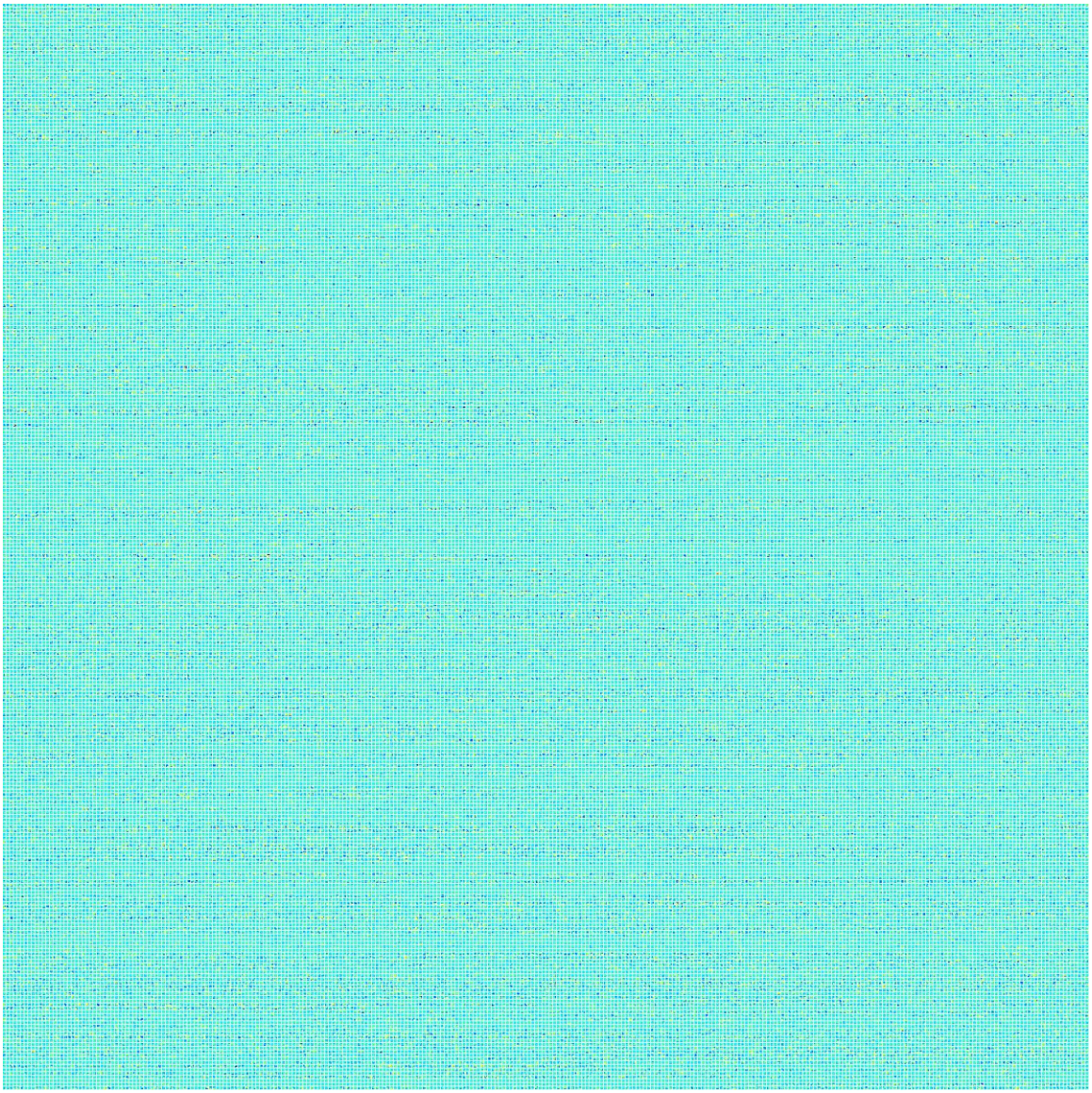


Figure A.9: In the trained AlexNet model, 384 filters of size of 3×3 pixels were applied and visualized for the third convolution layer of our network.

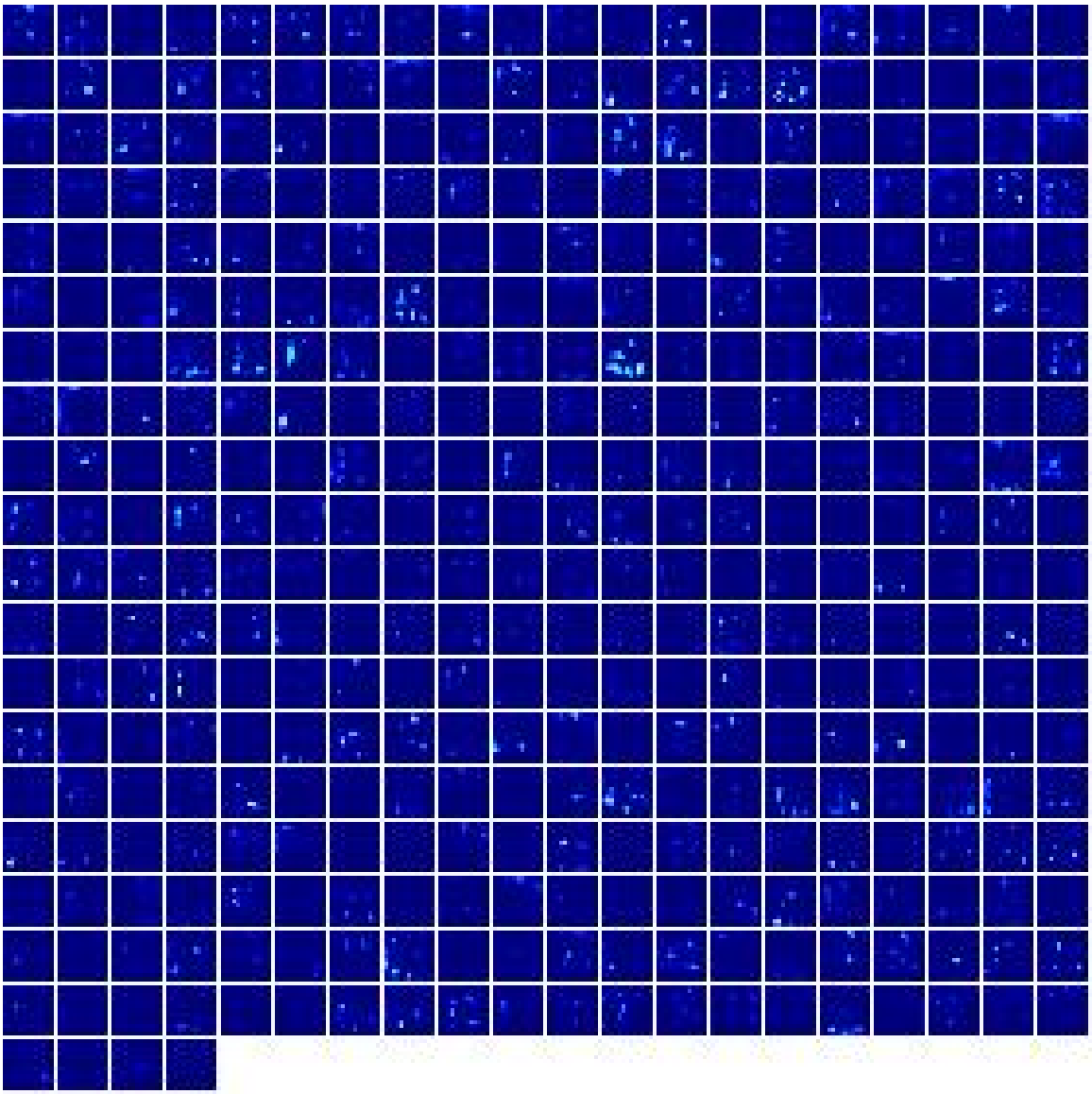


Figure A.10: In the third CONV layer of AlexNet in a given trained model 384 filters of 13×13 pixels were applied and visualized for a SMC patients sample.

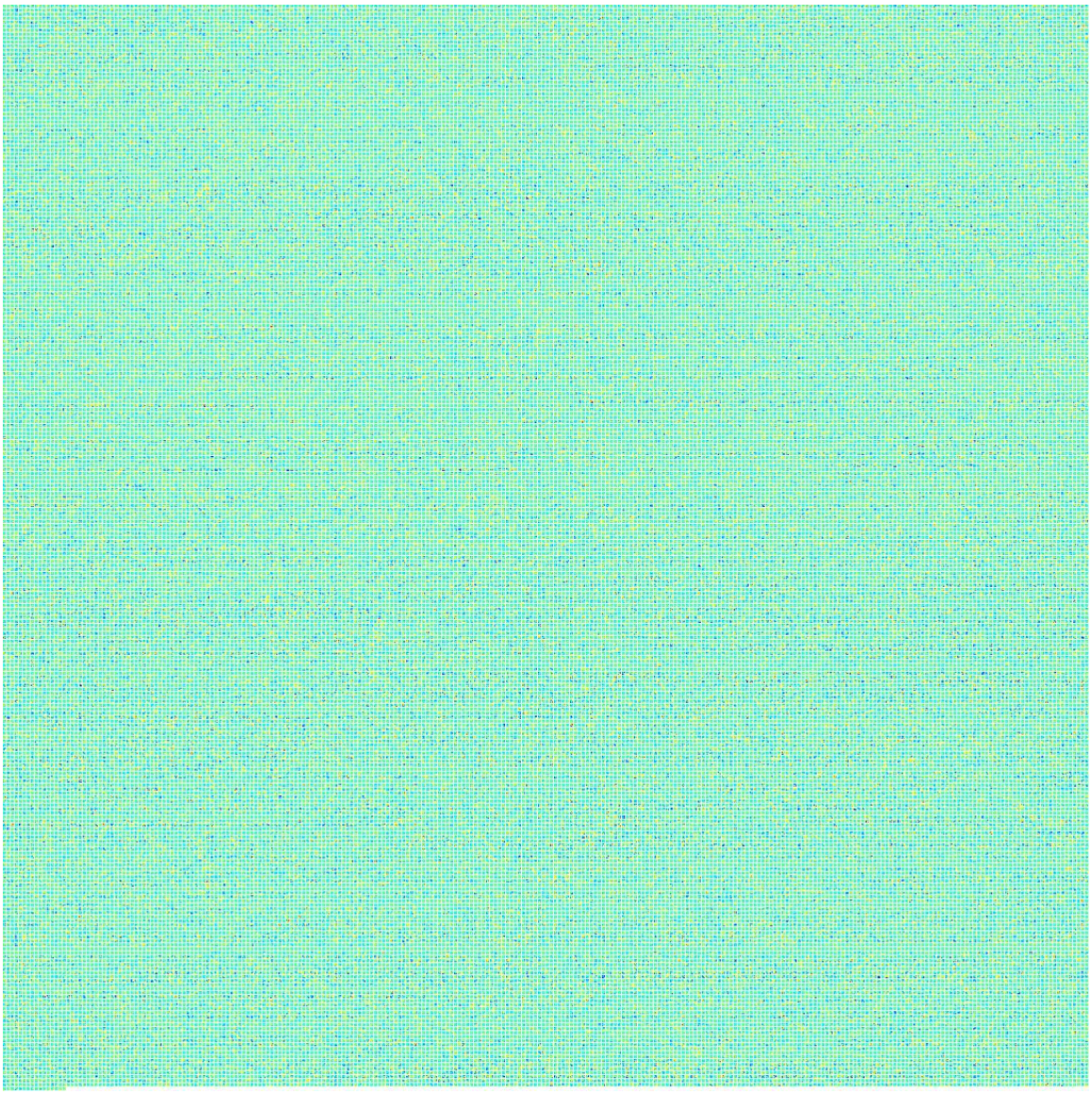


Figure A.11: In the trained AlexNet model, 384 filters of size of 3×3 pixels were applied and visualized for the fourth convolution layer of our network.

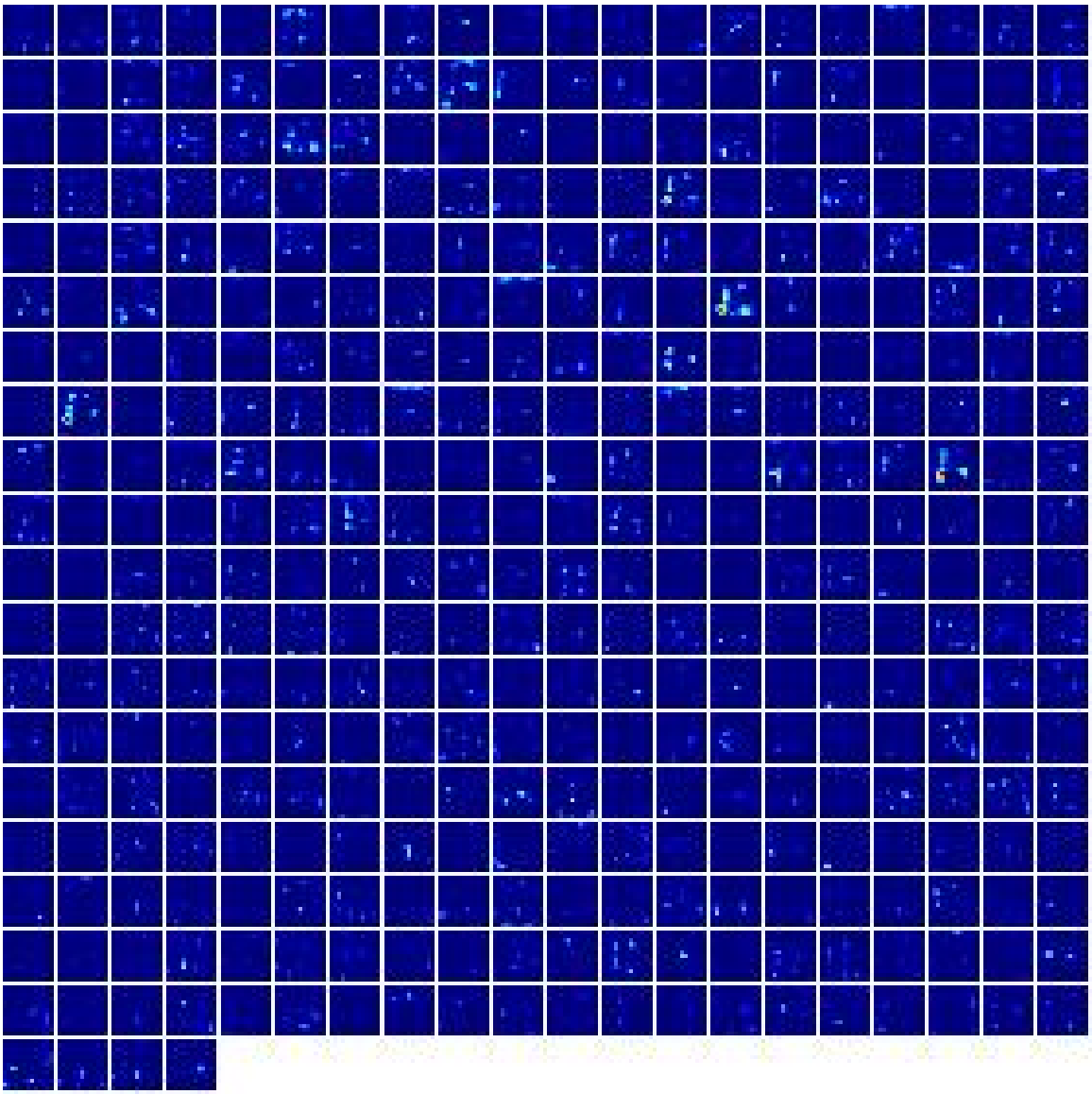


Figure A.12: In the third CONV layer of AlexNet in a given trained model 384 filters of 13×13 pixels were applied and visualized for a LMCI patients sample.

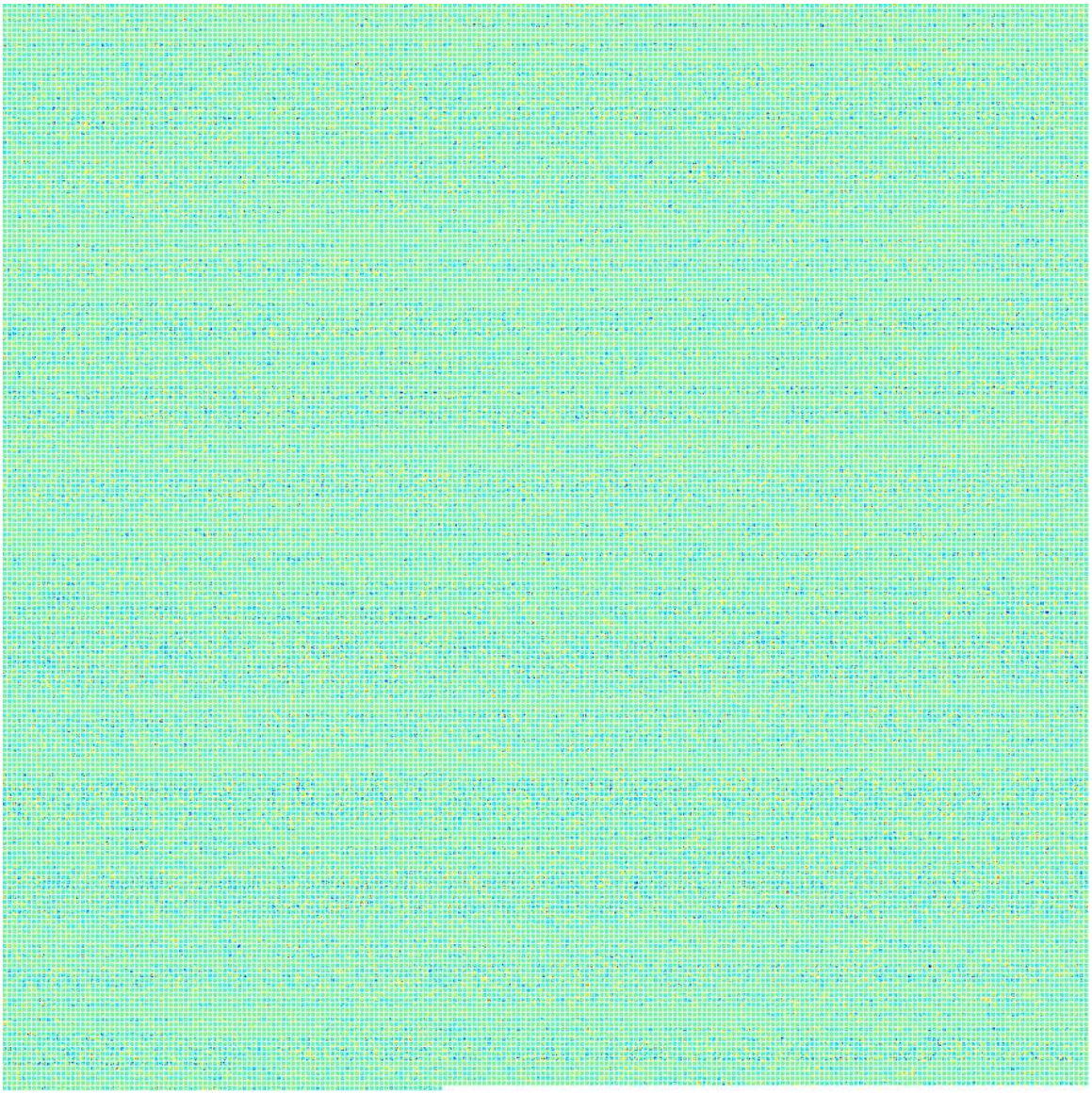


Figure A.13: In the trained AlexNet model, 256 filters of size of 3×3 pixels were applied and visualized for the fourth convolution layer of our network.

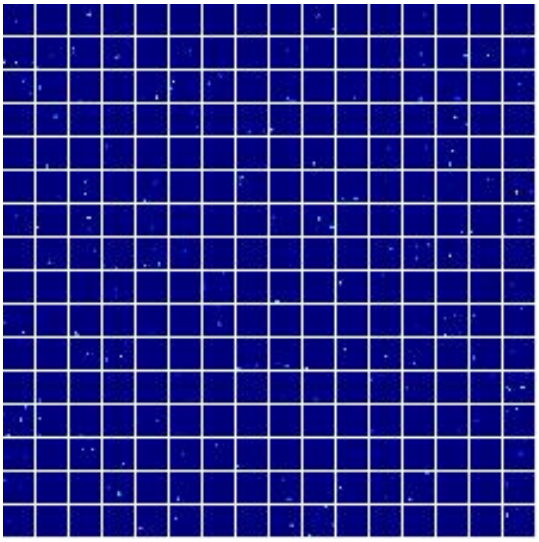


Figure A.14: In the third CONV layer of AlexNet in a given trained model 256 filters of 13×13 pixels were applied and visualized for a SMC patients sample.

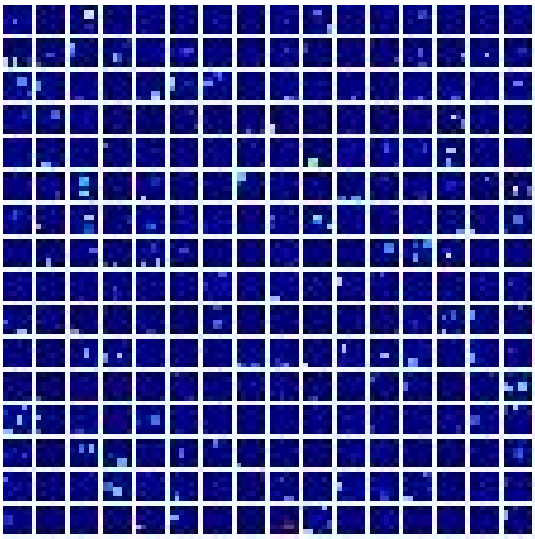


Figure A.15: The visualization of the third (last) pooling layer with 256 filters of size 6×6 pixels .

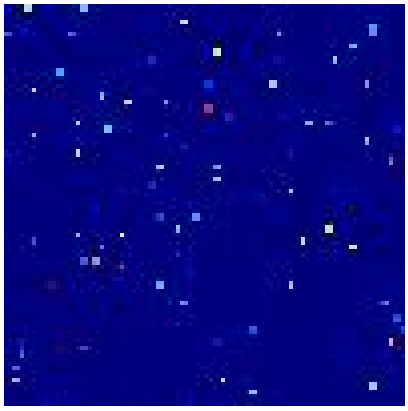


Figure A.16: The visualization of the six layer which is the first fully connected layer with 4096 nodes.

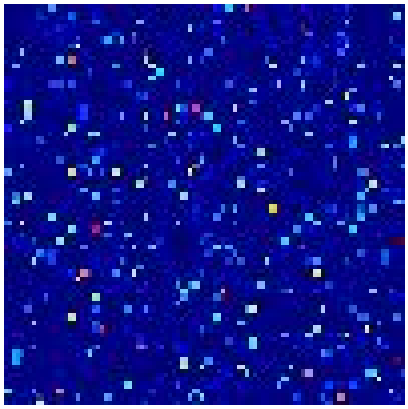


Figure A.17: The visualization of the seventh layer which is the second fully connected layer with 4096 nodes.

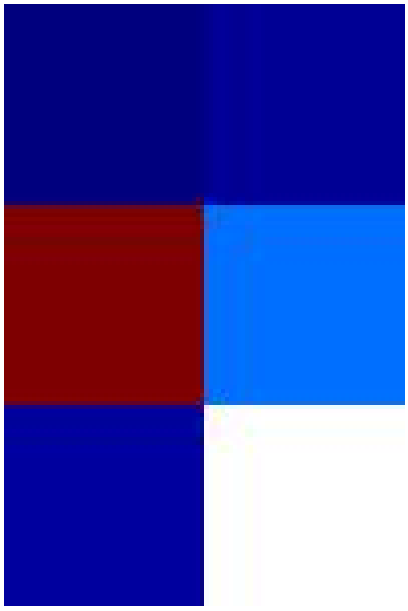


Figure A.18: The visualization of the eighth layer which is the third fully connected layer with 5 nodes.

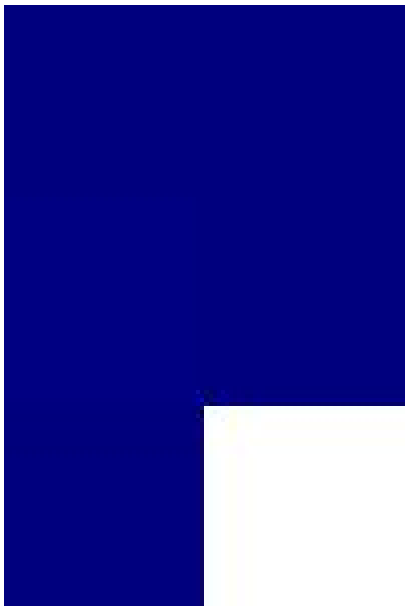


Figure A.19: The visualization of the last layer after applying softmax activation function.

Appendix B

Visualization results of different layers of AlexNet model for a sample of unsuccessful classification

In this Appendix we show the visualization of different layers of model for a patient in SMC class which was not classified into the correct category.

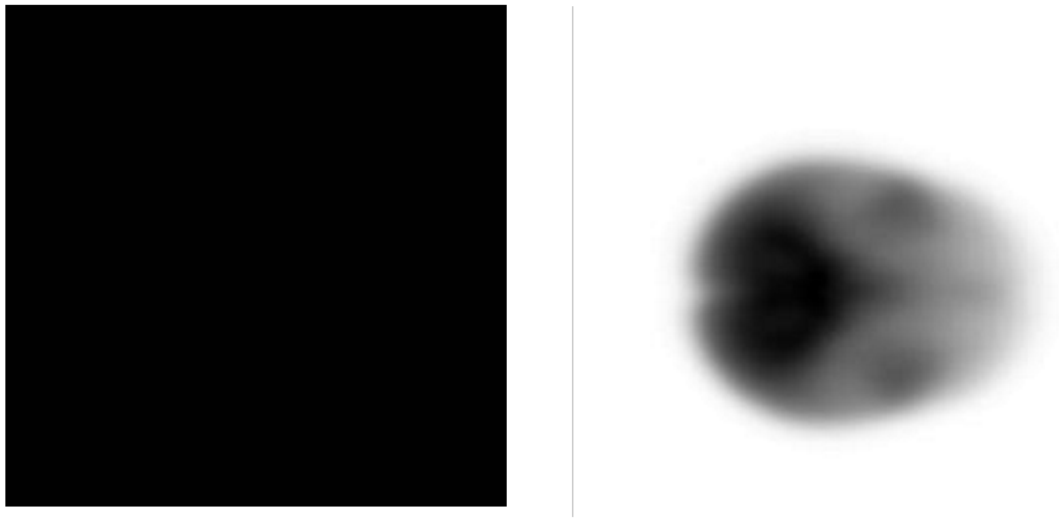


Figure B.1: Second slice of fMRI scan on time course of 136 for a SMC patient (left). The activation data as an input for the model. The size of filter applied in this part was 227×227

Class	prediction
AD	17.63%
EMCI	21.61%
LMCI	22.31%
NC	28.108%
SMC	10.27%

Table B.1: The prediction of the model for a sample image belong to SMC category.

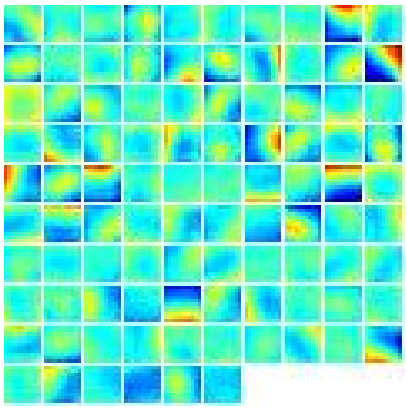


Figure B.2: In the trained AlexNet model, 96 filters of size of 11 × 11 pixels were applied and visualized for the first convolution layer of our network.

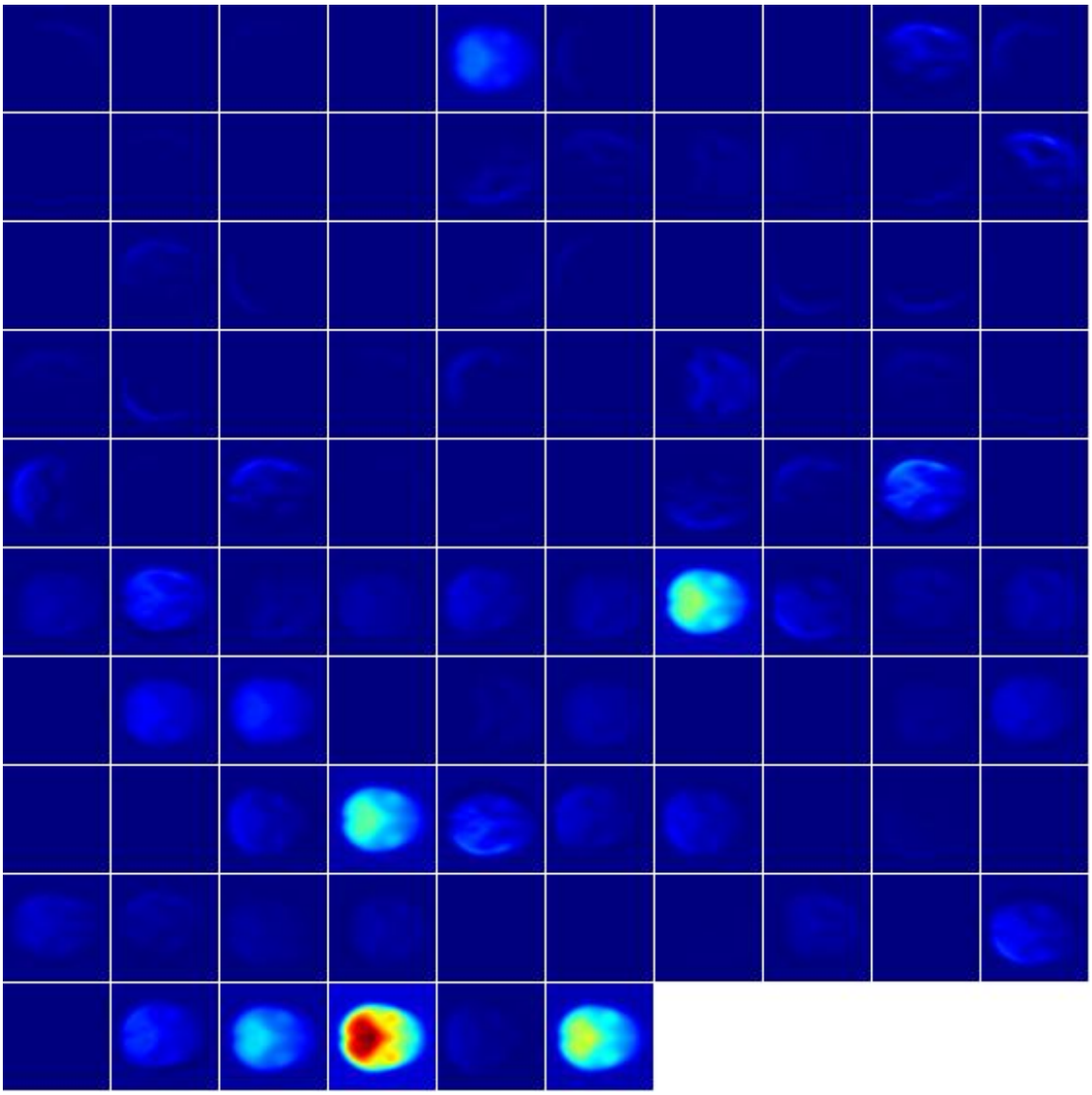


Figure B.3: In the first CONV layer of AlexNet in a given trained model 96 filters of 55×55 pixels were applied and visualized for a SMC patients sample. This images is a visualization of the first CONV layer after applying the activation function.

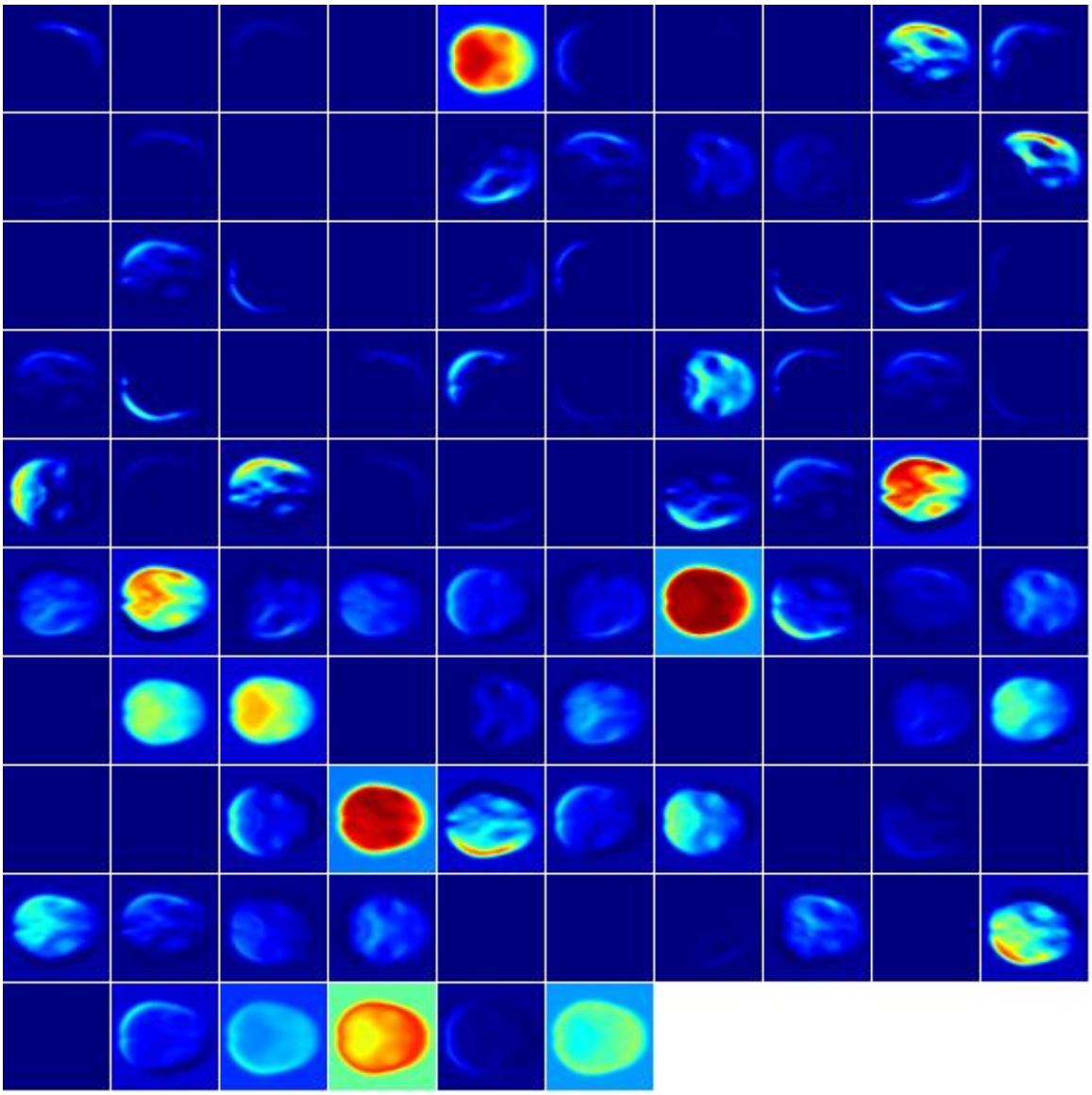


Figure B.4: The visualization of the first CONV layer after applying the activation function and normalization.

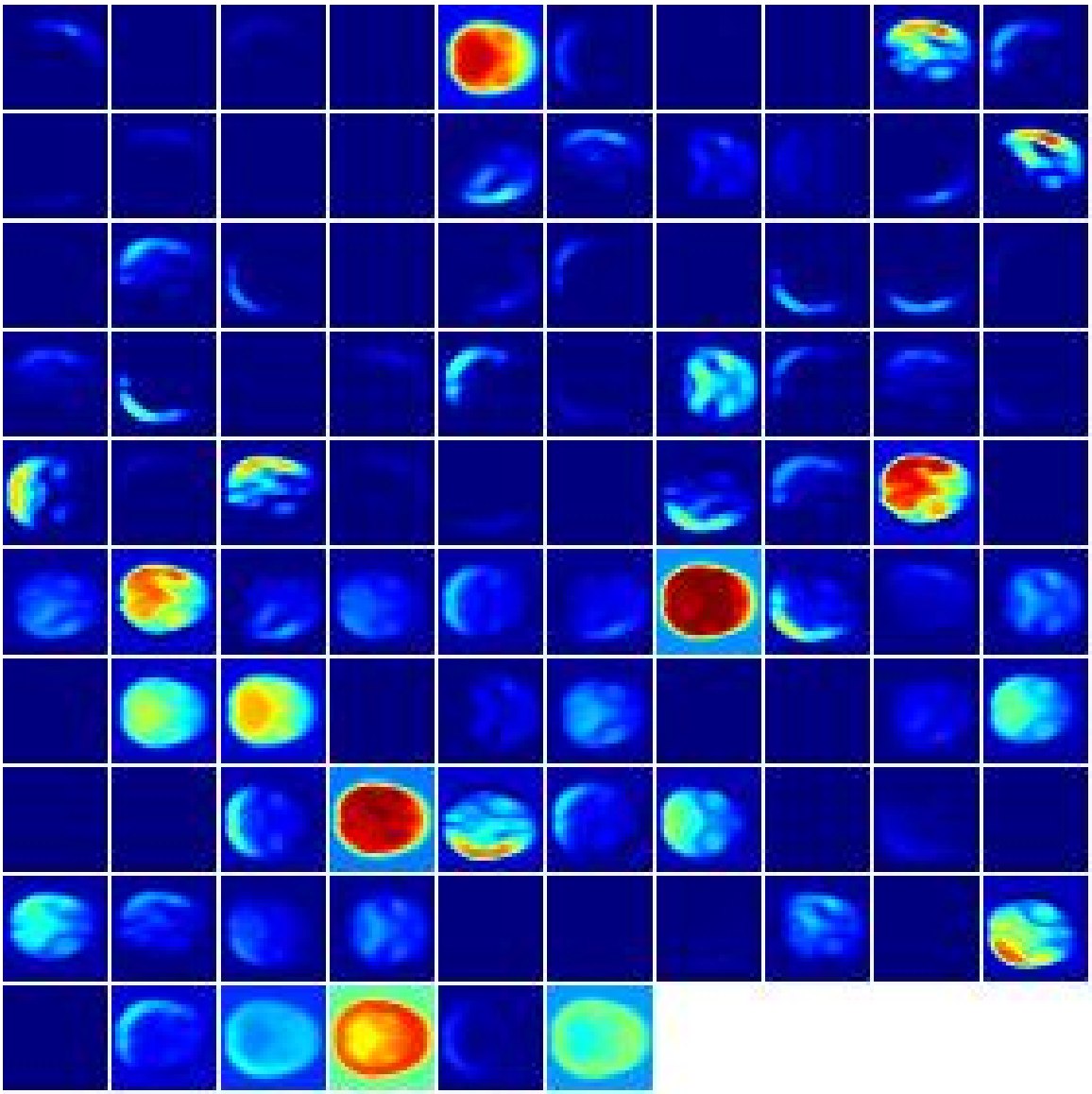


Figure B.5: The visualization of the first pooling layer with 96 filters of size 27 × 27 pixels
.

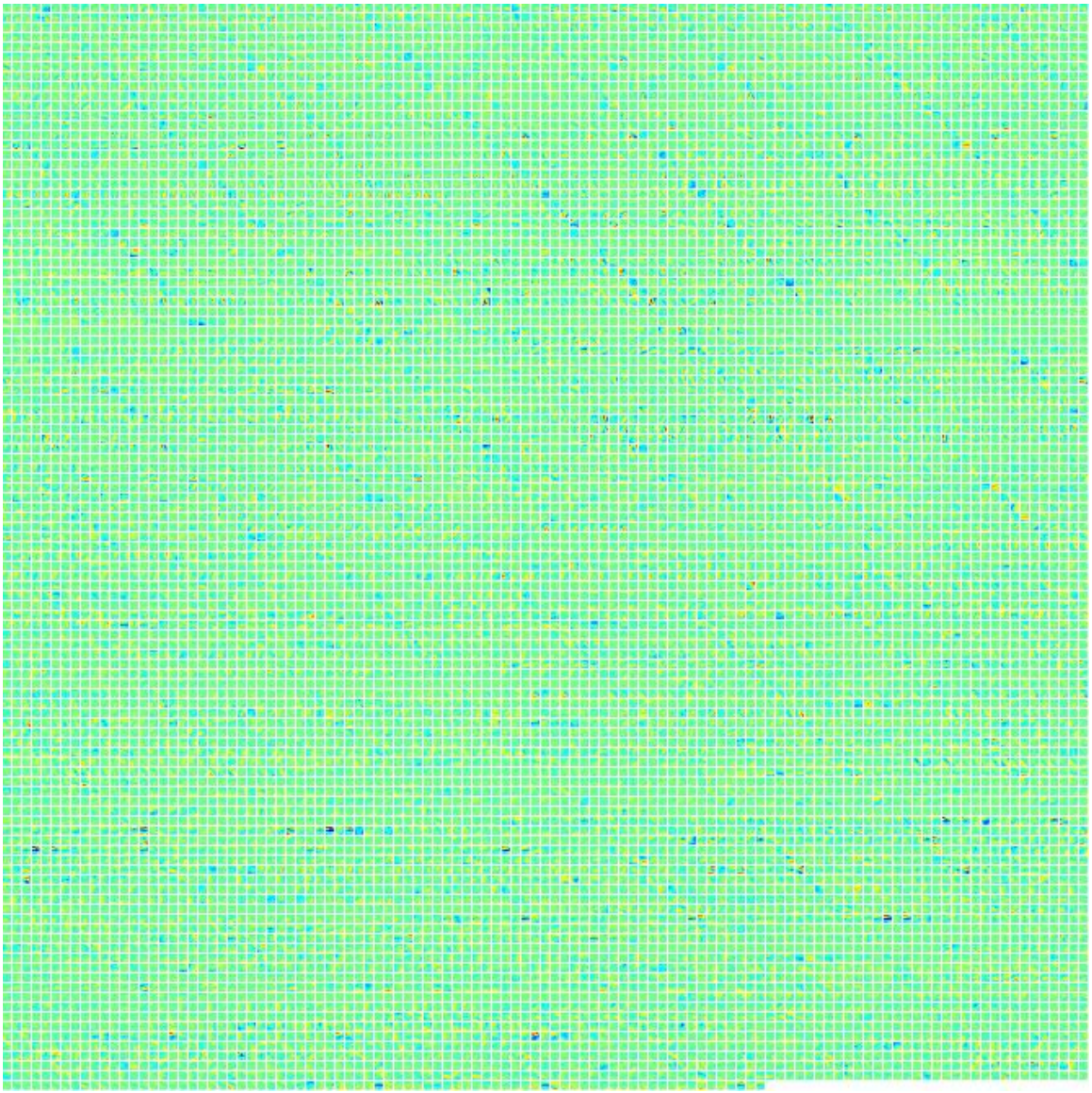


Figure B.6: In the trained AlexNet model, 256 filters of size of 5 × 5 pixels were applied and visualized for the second convolution layer of our network.

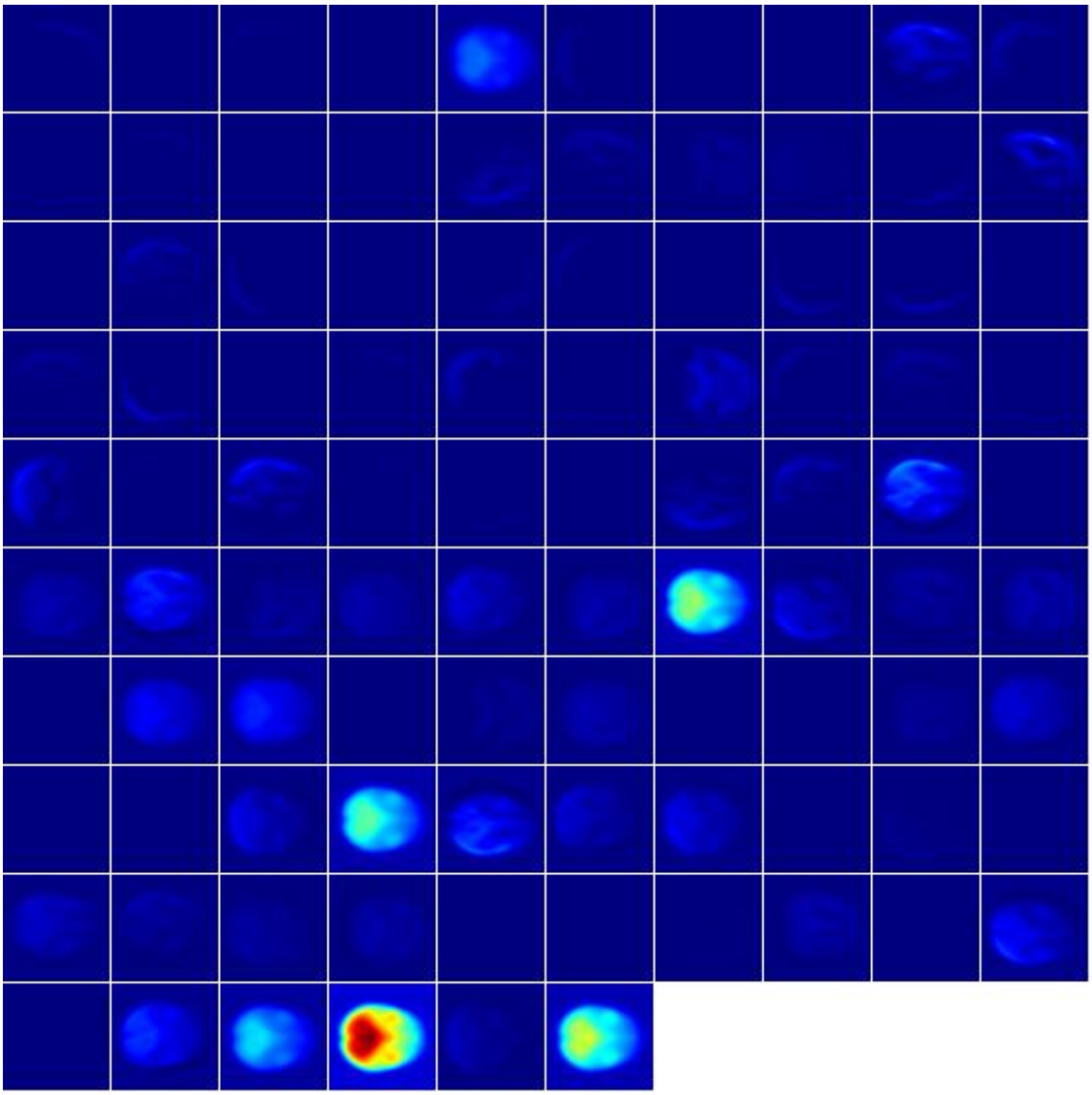


Figure B.7: In the second CONV layer of AlexNet in a given trained model 256 filters of 27×27 pixels were applied and visualized for a SMC patients sample.

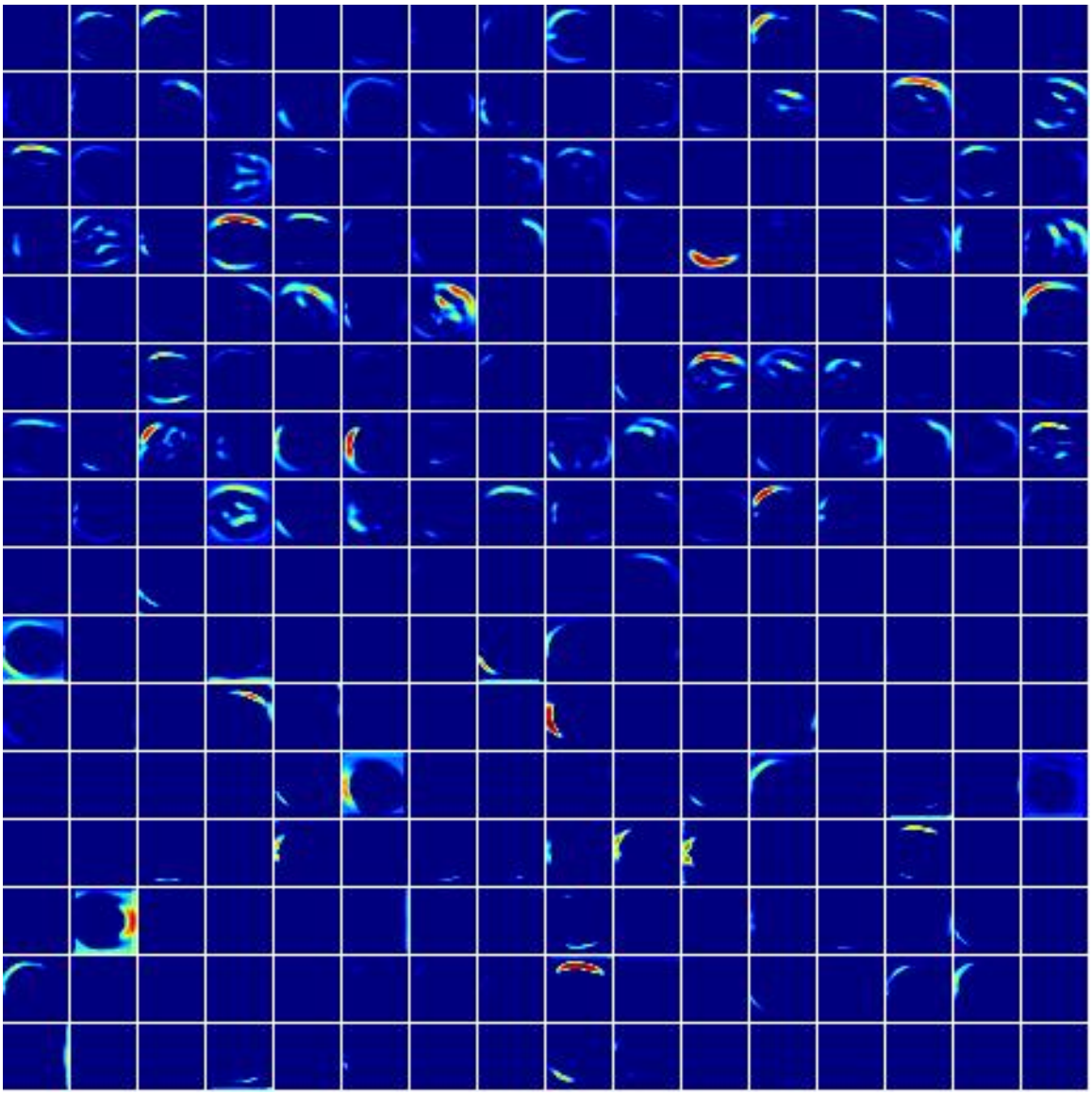


Figure B.8: The visualization of the second CONV layer after applying the activation function and normalization.

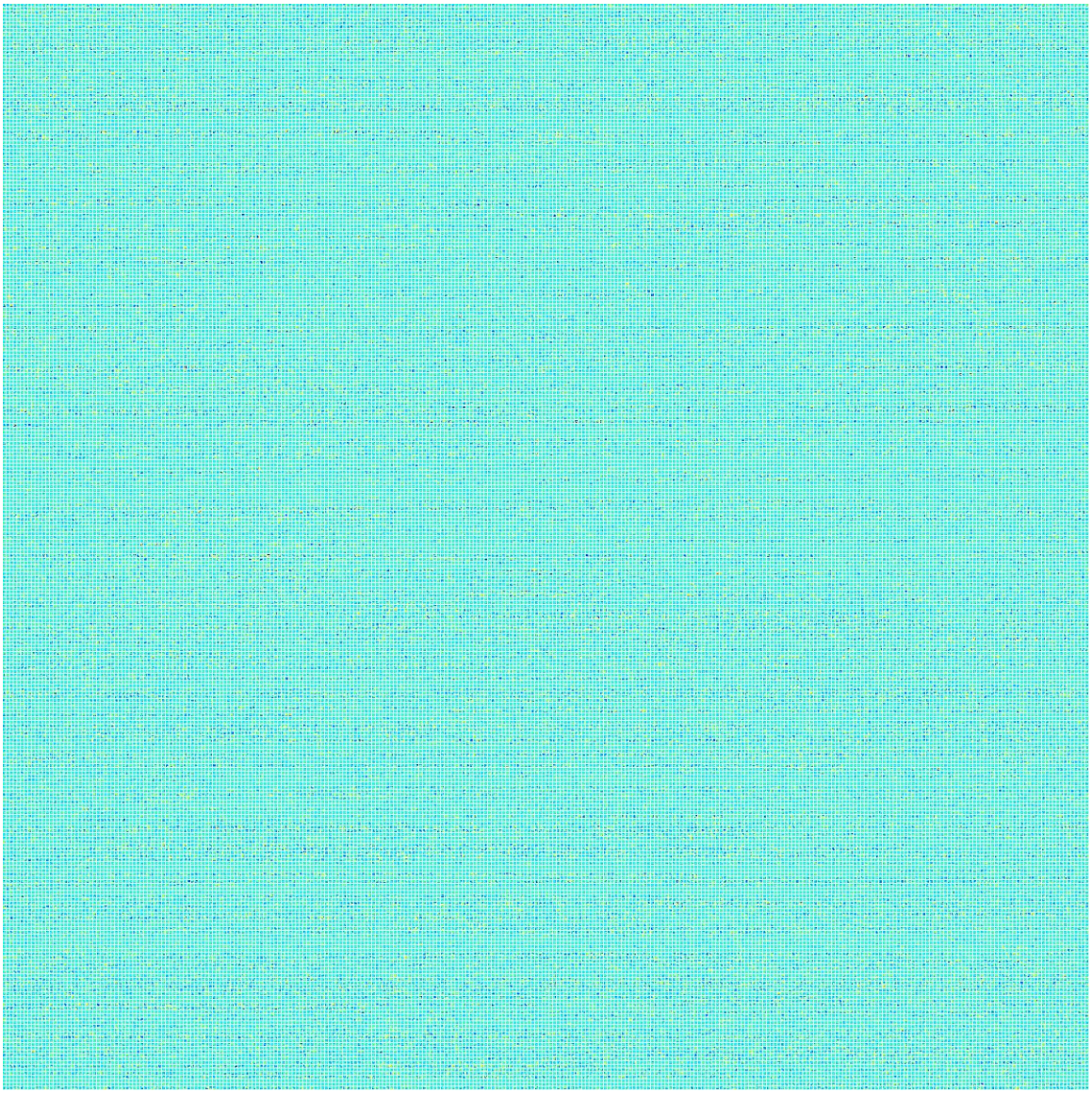


Figure B.9: In the trained AlexNet model, 384 filters of size of 3×3 pixels were applied and visualized for the third convolution layer of our network.

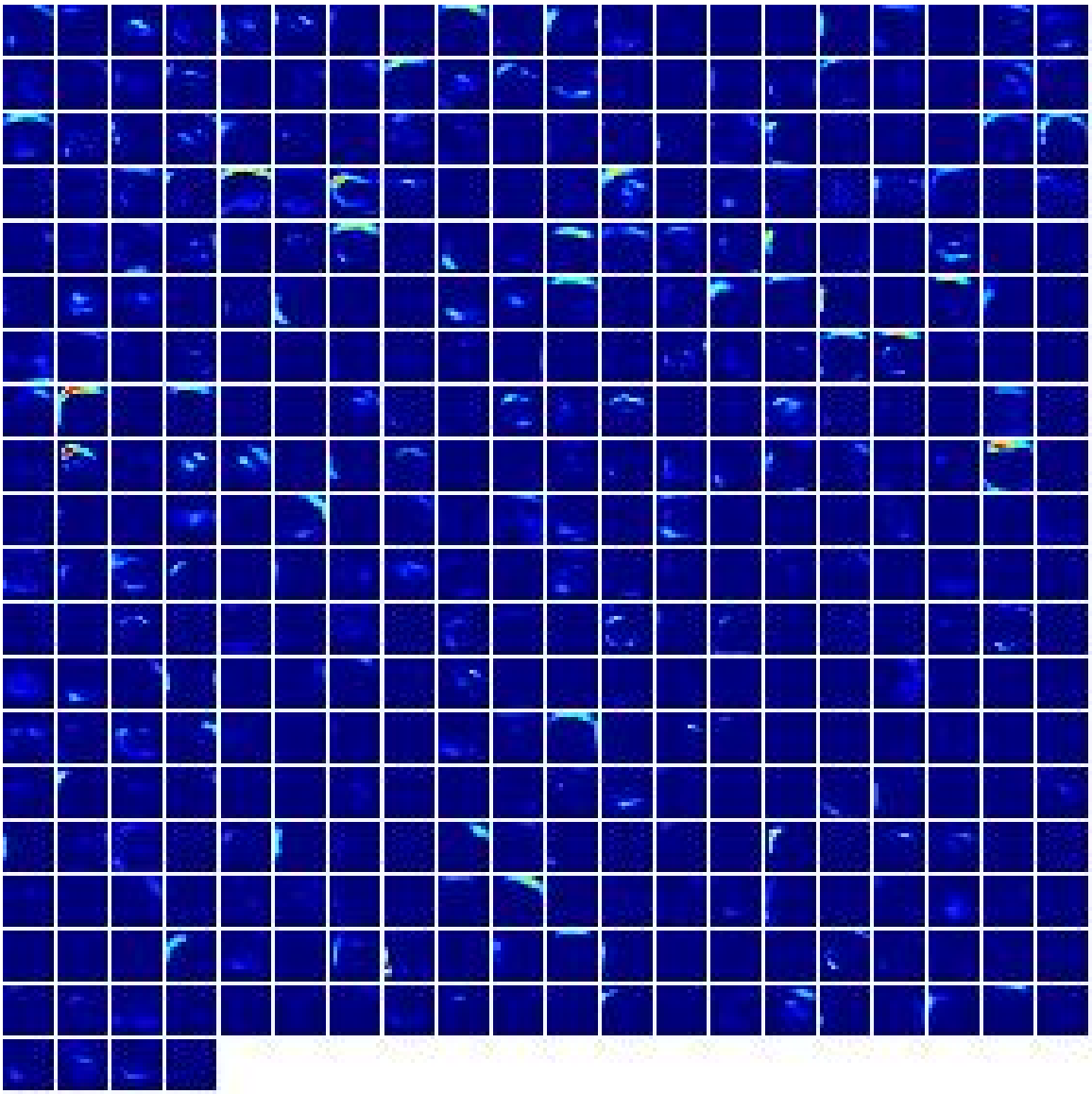


Figure B.10: In the third CONV layer of AlexNet in a given trained model 384 filters of 13×13 pixels were applied and visualized for a SMC patients sample.

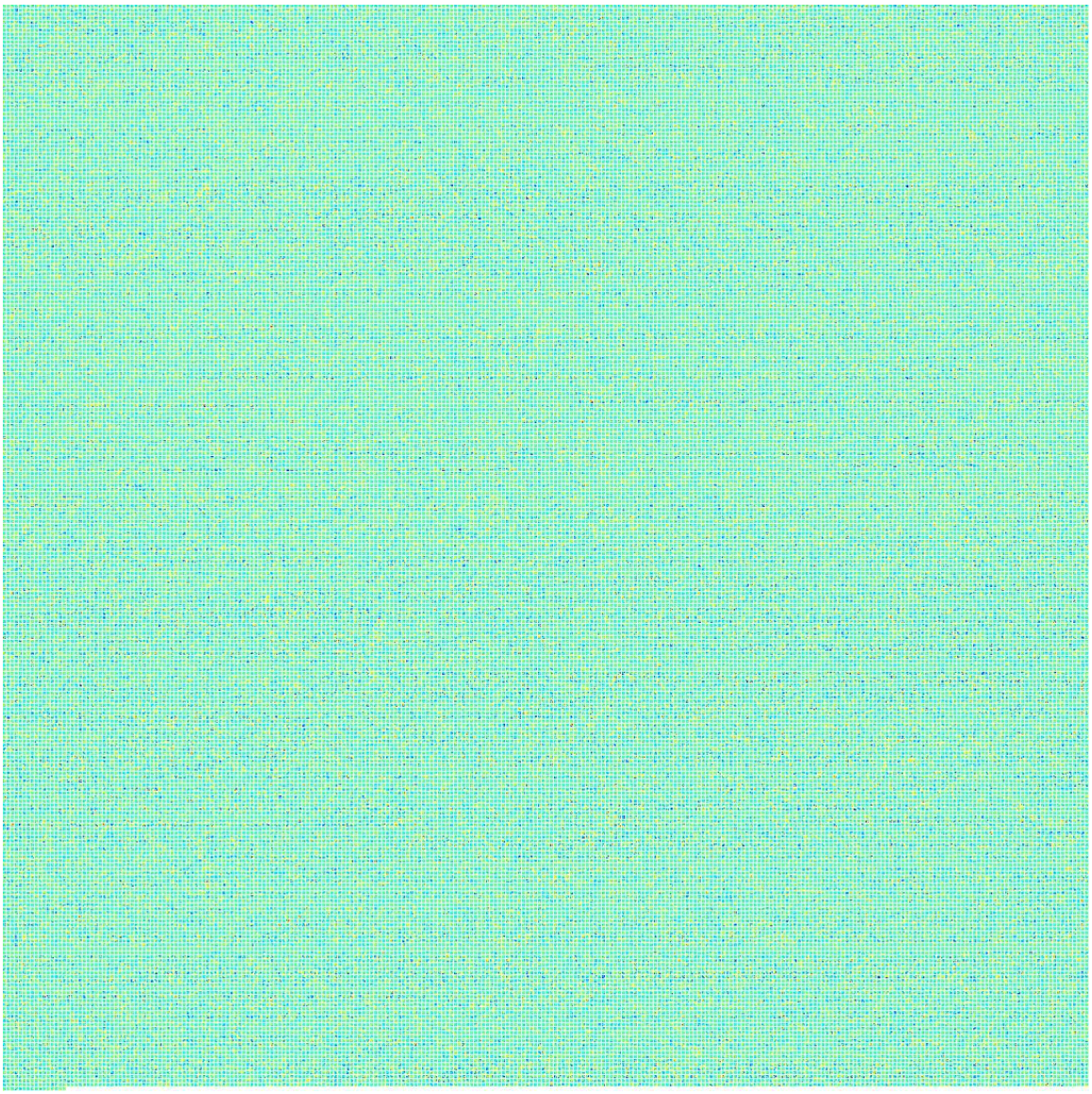


Figure B.11: In the trained AlexNet model, 384 filters of size of 3×3 pixels were applied and visualized for the fourth convolution layer of our network.

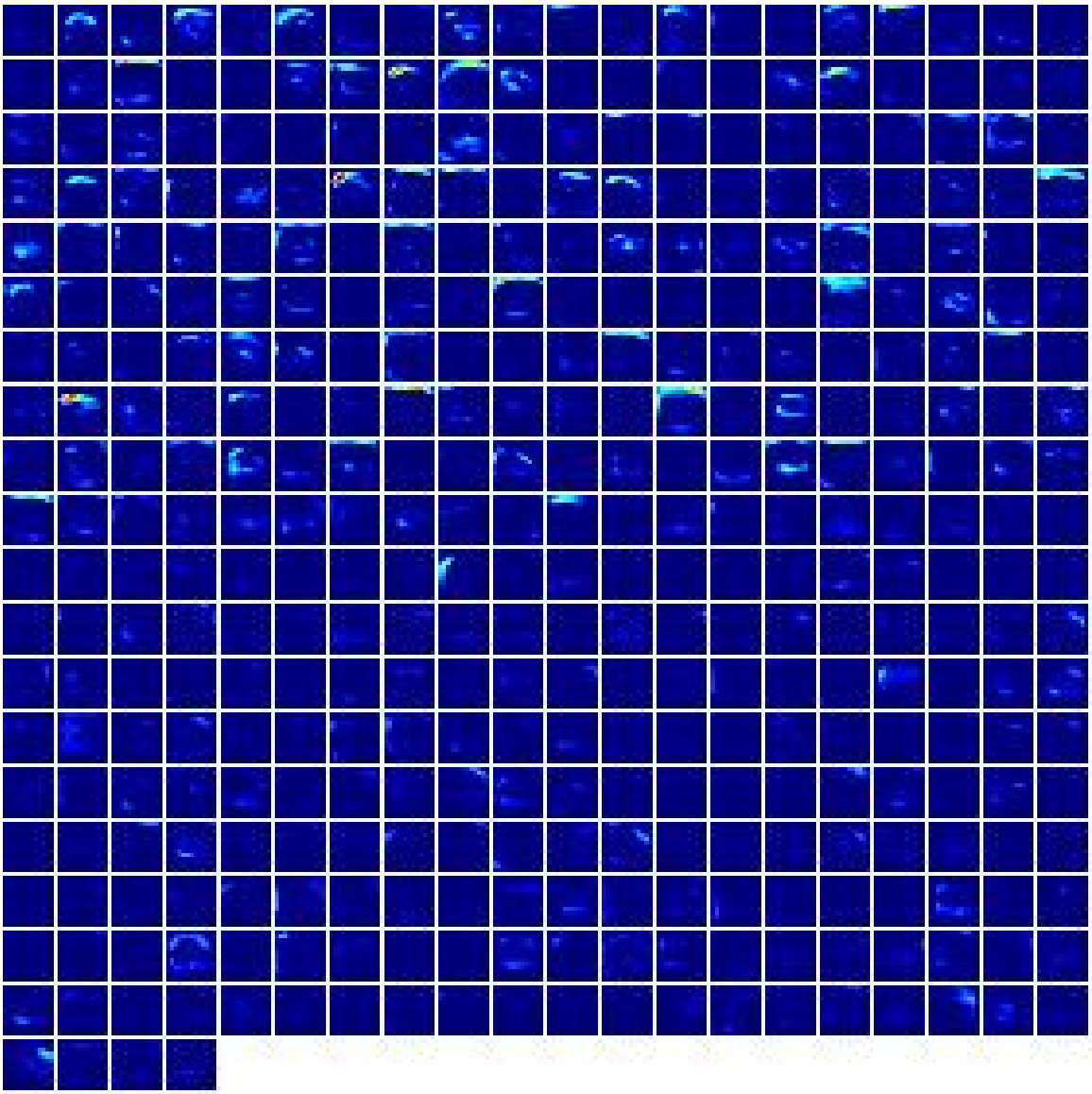


Figure B.12: In the third CONV layer of AlexNet in a given trained model 384 filters of 13×13 pixels were applied and visualized for a SMC patients sample.

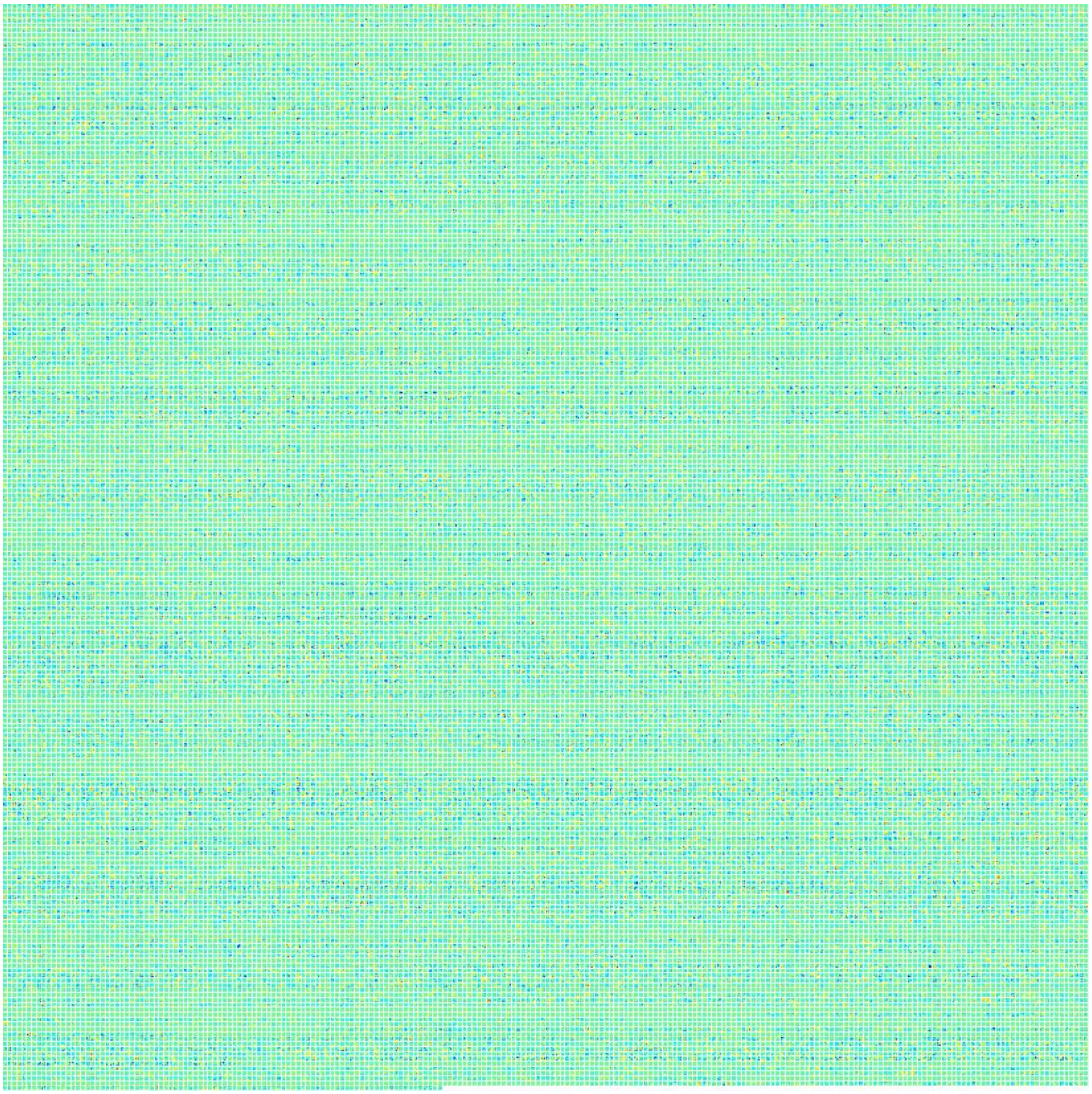


Figure B.13: In the trained AlexNet model, 256 filters of size of 3×3 pixels were applied and visualized for the fourth convolution layer of our network.

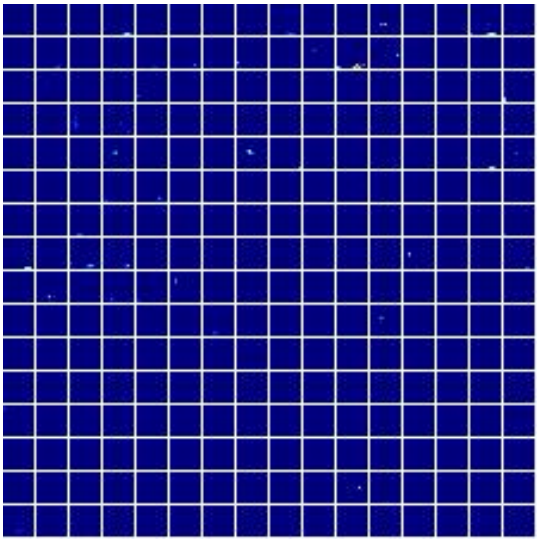


Figure B.14: In the third CONV layer of AlexNet in a given trained model 256 filters of 13×13 pixels were applied and visualized for a SMC patients sample.

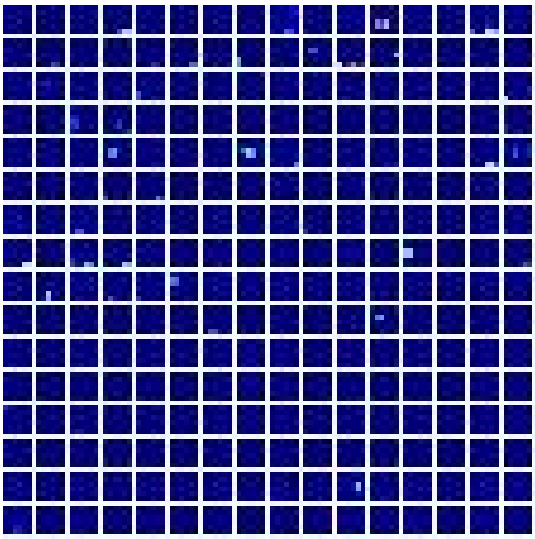


Figure B.15: The visualization of the third (last) pooling layer with 256 filters of size 6×6 pixels .

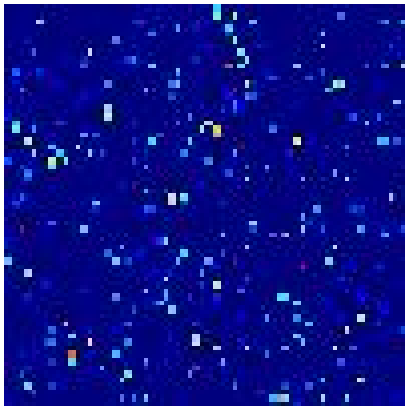


Figure B.16: The visualization of the six layer which is the first fully connected layer with 4096 nodes.

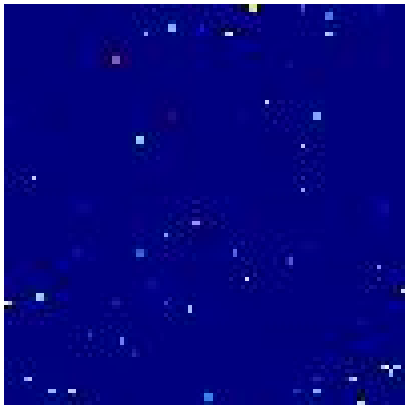


Figure B.17: The visualization of the seventh layer which is the second fully connected layer with 4096 nodes.

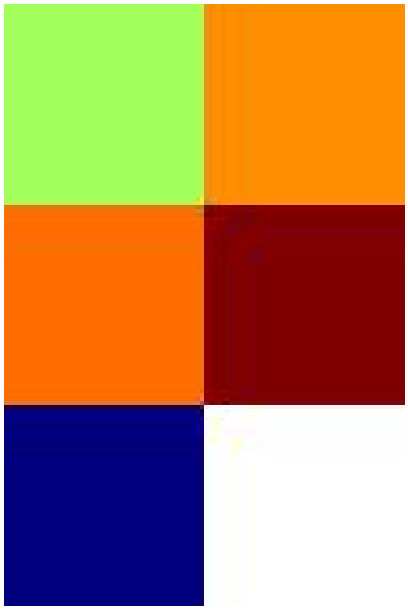


Figure B.18: The visualization of the eighth layer which is the third fully connected layer with 5 nodes.

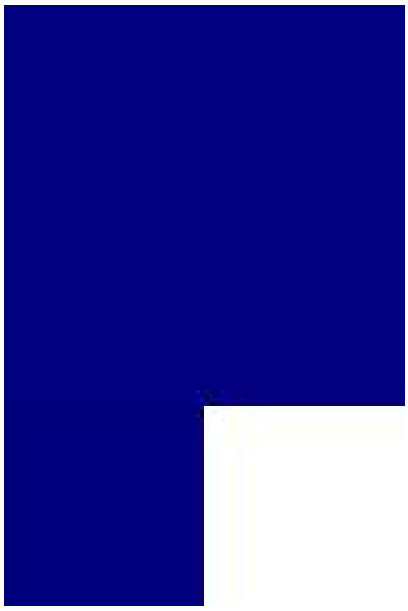


Figure B.19: The visualization of the last layer after applying softmax activation function.